ORDS:
An Introduction
How to Use this User Guide

This handbook accompanies the taught sessions for the course. Each section contains a brief overview of a topic for your reference. Most sections also contain one or more exercises.

Exercises are arranged as follows:

- A title and brief overview of the tasks to be carried out;
- A numbered set of tasks, together with a brief description of each;
- A numbered set of detailed steps that will achieve each task.

Some exercises, particularly those within the same section, assume that you have completed earlier exercises. Your teacher will direct you to the location of files that are needed for the exercises. If you have any problems with the text or the exercises, please ask the teacher or one of the demonstrators for help.

This book includes plenty of exercise activities – more than can usually be completed during the hands-on sessions of the course. You should select some to try during the course, while the teacher and demonstrator(s) are around to guide you. Later, you may attend follow-up sessions at ITLP called Course Clinic where you can continue work on the exercises, with some support from IT teachers. Other exercises are for you to try on your own, as a reminder or an extension of the work done during the course.

Text Conventions

A number of conventions are used to help you to be clear about what you need to do in each step of a task.

- In general, the word **press** indicates you need to press a key on the keyboard. **Click, choose** or **select** refer to using the mouse and clicking on items on the screen. If you have more than one mouse button, click usually refers to the left button unless stated otherwise.
- Names of keys on the keyboard, for example the Enter (or Return) key are shown like this: **ENTER**.
- Multiple key names linked by a + (for example, CTRL+Z) indicate that the first key should be held down while the remaining keys are pressed; all keys can then be released together.
- Words and commands typed in by the user are shown **like this**.
- Labels and titles on the screen are shown **like this**.
- Drop-down menu options are indicated by the name of the options separated by a vertical bar, for example **File|Print**. In this example you need to select the option **Print** from the **File** menu or tab. To do this, click when the mouse pointer is on the **File** menu or tab name; move the pointer to **Print**; when **Print** is highlighted, click the mouse button again.
- A button to be clicked will look **like this**.
- The names of software packages are identified **like this**, and the names of files to be used **like this**.
Software Used

Firefox or Chrome web browser
Microsoft Excel

Files Used

Clubs_database.sql

Revision Information

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Author(s)</th>
<th>Changes made</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>November 2014</td>
<td>James A J Wilson &amp; Meriel Patrick</td>
<td>Created</td>
</tr>
<tr>
<td>1.1</td>
<td>November 2014</td>
<td>Meriel Patrick</td>
<td>Minor changes to exercises</td>
</tr>
<tr>
<td>1.2</td>
<td>February 2015</td>
<td>Meriel Patrick</td>
<td>Minor updates</td>
</tr>
<tr>
<td>1.3</td>
<td>May 2015</td>
<td>Meriel Patrick</td>
<td>Minor updates and new section added</td>
</tr>
<tr>
<td>1.4</td>
<td>November 2015</td>
<td>Meriel Patrick</td>
<td>Updates to reflect new functionality</td>
</tr>
</tbody>
</table>

Copyright

This document and all images therein is copyright of the University of Oxford. The Oxford University logo and crest is copyright of Oxford University and may only be used by Oxford University members in accordance with the University’s branding guidelines.
Contents

1 Introduction ............................................................................................................. 1
  1.1. What you should already know ...................................................................... 1
  1.2. What will you learn? ......................................................................................... 1
  1.3. What is the ORDS? ............................................................................................ 2
  1.4. Why use the ORDS? ......................................................................................... 2
  1.5. Where do I find the ORDS? .............................................................................. 2

2 Getting started with the ORDS .............................................................................. 4
  2.1. Registering to use the ORDS ........................................................................... 4
    2.1.1. Members of the University of Oxford and virtual card holders .................. 4
    2.1.2. ORDS users from outside the University of Oxford .................................... 5
    2.1.3. Registration .................................................................................................. 6
  2.2. Your account ...................................................................................................... 6
  2.3. Creating a new project ....................................................................................... 7
    2.3.1. Trial projects and full projects .................................................................... 9
  2.4. Project members and their permissions ............................................................ 10
    2.4.1. Adding and removing project members ...................................................... 11
  2.5. Project audit record ........................................................................................... 11
  2.6. Editing project details ....................................................................................... 12
  Exercise 1: Registering to use the ORDS ............................................................... 12
  Exercise 2: Creating a new project .......................................................................... 12

3 Creating databases .................................................................................................. 13
  3.1. Importing an existing database ....................................................................... 15
  3.2. Importing data in spreadsheets ....................................................................... 16
  3.3. Using the database schema editor ..................................................................... 18
    3.3.1. Adding and removing tables ....................................................................... 19
    3.3.2. Adding new fields ....................................................................................... 21
    3.3.3. Data types .................................................................................................... 21
    3.3.4. Other ORDS field options ......................................................................... 24
    3.3.5. Relationships between tables ...................................................................... 26
    3.3.6. Saving changes to a database structure ...................................................... 31
    3.3.7. Common problems when changing database structures .......................... 32
  3.4. Milestone and test versions of databases ......................................................... 32
    3.4.1. Creating and deleting a milestone or test version of a database ............... 33
  3.5. Exporting databases .......................................................................................... 33
Exercise 3: Importing a database file ........................................ 34
Exercise 4: Viewing and editing the database structure .................. 34

4 Editing data ............................................................................ 35
  4.1. The ORDS editing screen .................................................. 35
  4.2. Editing data in linked tables ............................................. 37
  4.3. Filtering records ................................................................ 39
  4.4. Adding and deleting records ............................................. 44
  4.5. Adding new records in bulk ............................................. 45
Exercise 5: Viewing and editing data ........................................... 45

5 Querying a database .............................................................. 46
  5.1. The Query Builder ............................................................ 46
  5.2. SQL queries ..................................................................... 51
    5.2.1. SQL basics ............................................................... 52
  5.3. Exporting query results .................................................... 53
  5.4. Saving query results as datasets ...................................... 53
Exercise 6: Querying a database .................................................. 54

6 Connecting to the ORDS from third-party software and websites .......................................................... 56

7 What next? .............................................................................. 57
  7.1. Database design courses ................................................... 57
  7.2. Relational database management systems courses ............. 57
  7.3. Course Clinic .................................................................... 57
  7.4. Further help ....................................................................... 57
  7.5. Downloadable course materials and more – the ITLP Portfolio ... 57
  7.6. Useful websites ................................................................... 58

Appendix: Student Exercises ...................................................... 59
Exercises

Exercise 1  Registering to use the ORDS................................. 59
Exercise 2  Creating a new project........................................ 61
Exercise 3  Importing a database file...................................... 62
Exercise 4  Viewing and editing the database structure .......... 64
Exercise 5  Viewing and editing data........................................ 67
Exercise 6  Querying a database.............................................. 71
Exercise 7  Creating a new database from scratch............... 75
1 Introduction

Welcome to the Online Research Database Service (ORDS) training course!

This booklet accompanies the course delivered by Oxford University’s IT Learning Programme. Although the exercises are clearly explained so that you can work through them independently, you will find that it will help if you also attend the taught session where you can get advice from the teacher, demonstrator(s) and even each other.

If at any time you are not clear about any aspect of the course, please make sure you ask your teacher or demonstrator for some help. If you are away from the class, you can get help by email from your teacher or from help@it.ox.ac.uk.

1.1. What you should already know

No previous knowledge of the ORDS is expected. If you are already familiar with relational databases then you will find the examples used in this course easier to understand; those with no experience should be able to acquire a basic understanding of how relational databases work during the session, although it is recommended that they follow this up by attending the course Databases: Concepts of database design.

We will assume that you are familiar with opening files from particular folders and saving them, perhaps with a different name, back to the same or a different folder.

The computer network in our teaching rooms may differ slightly from that which you are used to in your College or Department; if you are confused by the differences please ask for help from the teacher or demonstrator(s).

1.2. What will you learn?

This course will help you learn to use the ORDS to create, edit, and share databases.

In this session we will cover the following topics:

- Setting up projects, adding collaborators, and assigning permissions
- Creating new databases from scratch
- Importing existing databases and data
- Designing and restructuring databases
- Viewing and editing data
- Querying data and saving the results
- Exporting datasets and databases
1.3. What is the ORDS?

The Online Research Database Service (ORDS) is a free, centrally supported service for researchers based at the University of Oxford, and their collaborators. It was developed in-house by IT Services and is supported by the Research Support Team and Systems Development Team. It offers researchers a place to host relational databases, along with online tools to edit and share their data.

1.4. Why use the ORDS?

There are a number of reasons why the ORDS might be preferred to other relational database management systems.

- All data in the ORDS is stored on university infrastructure and backed-up nightly, reducing the risk of data loss
- Enables easy collaboration both with Oxford colleagues and partners at other institutions
- No need to set up servers oneself for hosting online databases
- No confusion over different versions of the same database – all collaborators edit the current up-to-date version by default
- Data can easily be shared either between named project associates or with the wider public
- Easy to make subsets of databases public whilst keeping the full database private
- Shared datasets may be ‘static’ (fixed at the point of publication) or ‘dynamic’ (updating as the underlying database is edited)
- Pre-existing Access databases can be uploaded for wider sharing or continued development
- Data in spreadsheets can be uploaded and stitched together to create full relational databases
- Databases can be exported in an open format (PostgreSQL) and imported into other systems if circumstances change – users are not tied in
- Simple data editing interface
- In-house support
- Free (up to 20 GB)

Development of the ORDS is on-going. Requests for additional features may be sent to ords@it.ox.ac.uk, where they will be reviewed and, if deemed appropriate, prioritized for future development.

If you are unsure whether the ORDS would be the most appropriate solution for your research, the support team will be happy to advise.

1.5. Where do I find the ORDS?

The ORDS, as its name suggests, is an online service. You can access the ORDS from https://app.ords.ox.ac.uk/. It should work in any web browser, although testing has focussed on Firefox and Chrome. Supplementary information about the service, along with documentation and how-to guides are available from http://ords.ox.ac.uk/.
Offline editing of data hosted in the ORDS is possible using third-party database management software such as Microsoft Access, although users will need to connect to the ORDS database they wish to edit before going offline. Data will then need to be re-uploaded to the ORDS once the user has an internet connection again. The process for uploading data edited offline is only semi-automated at present, so users wishing to edit data offline should contact ORDS support staff in advance via ords@it.ox.ac.uk to indicate that this is what they are intending to do.
2 Getting started with the ORDS

Before you can start to use the ORDS you will need to complete the online registration process. Please note that logging in to the ORDS and registering to use the ORDS are separate processes.

- Logging in to the ORDS is the process of gaining access to your ORDS account by entering your username and password. You will do this every time you use the ORDS system.
- Registering to use the ORDS is a one-off process which adds your details to the list of registered ORDS users. You need to do this only on the first occasion that you use the ORDS.

2.1. Registering to use the ORDS

To start using the ORDS system, you will need to log in using a username and password and then register. The credentials you use will vary depending on your status.

2.1.1. Members of the University of Oxford and virtual card holders

For members of the University of Oxford, logging in to the ORDS is done via the SSO (single sign-on) service. This is the same username and password that you use to access your Nexus email account.

The main ORDS home page is: https://app.ords.ox.ac.uk/. This is the version you should use if you wish to use ORDS for your research, or to try the system out in your own time.

However, if you are using the system during an ORDS training course at IT Services, please use the staging environment: https://staging.ords.ox.ac.uk/. This is a second version of the ORDS system used for training and testing. It works in exactly the same way as the main version.

To log in, click the Log in using SSO button.
You will be prompted to enter your University of Oxford SSO username and password. When you have done this, click the button, then click the button on the confirmation page that follows.

If this is the first time you have used the ORDS system, you will now be prompted to register. If you have already registered, you will be taken to the ORDS welcome page.

2.1.2. ORDS users from outside the University of Oxford

The ORDS is also accessible to researchers from outside Oxford who have been invited to join a project. Non-Oxford users cannot be an ‘owner’ of an ORDS project, but they can otherwise participate in project with the same permissions as members of the University.

If the non-Oxford user is a member of a university or other institution that is part of the UK Access Management Federation, then they will be able to log in using their institution’s equivalent of an SSO username and password, by clicking the button and selecting their institution from the list.

If the non-Oxford user does not belong to an institution that is a member of the UK Access Management Federation then they will need to request a Virtual University Card before they are able to log in. A request for an application form should be sent to ords@it.ox.ac.uk. It can take a few weeks to process an application for a Virtual Card.

Note that the UK Access Federation, despite its name, includes several non-UK educational institutions.
### 2.1.3. Registration

The first time you log in to the ORDS, you will be presented with the registration page. This is where you can choose the email address to which messages from the ORDS will be sent, and how you wish to be referred to within the system.

If you logged in with your SSO account you should find that your name and email address have been completed automatically. Check that they are appearing correctly and edit them if necessary.

If you have been invited to join an ORDS project by a colleague, you should register with the email address they sent the invitation to, otherwise the system will not recognize that you should be added as a member of the project you were invited to join. (If you wish, you can change your email address later.)

When you are happy that your name and email address are correct, click the button. This will trigger a confirmation email, which will be sent to the email address you have just entered. That email will contain a link that will allow you to verify your ORDS account.

If you do not receive the email, it may be worth checking your spam or junk mail folder (to ensure that you can receive notifications from ORDS in the future, you may also wish to add ords@it.ox.ac.uk to your email program’s whitelist, or list of approved senders).

The confirmation email will normally arrive within a few minutes. If you are trying to register during the ITLP course, let the course leader know if this does not happen. Otherwise, if you have not received the message after an hour or so, please contact the ORDS help desk by emailing ords@it.ox.ac.uk.

### 2.2. Your account

Once you have logged in and registered, you will be taken to your project list, which lists all the projects of which you are currently a member. If you are a
member of the University of Oxford, you will also see the Create new project button. Before learning how to create a new project, however, it is worth taking a quick look at your account page.

To access your account information, click the Account info button or on your name in the ORDS title bar.

Your account screen is where you can change the name by which you are known within the ORDS, edit your email address, and indicate that you would like a project to be accessible via an ODBC connection. ODBC connections are used if you wish to edit or display ORDS data using third-party software or via a website. This is discussed in a little more detail in section 6.

To return to your project list, click your browser’s Back button, or the ORDS logo in the title bar at the top-left of your screen. Clicking the logo always returns you to your project list, or to the welcome page if you are not logged in.

2.3. Creating a new project

Only registered ORDS users who are members of the University of Oxford can create new projects. A project may be thought of as an electronic container for one or more databases. Each project has a project owner, who is the user with primary responsibility for it. By default, the user who creates a project is its owner.

To create a project, click the Create new project button. You will be taken to the Create New Project form, where you are prompted to enter some information about the project.
As a minimum, you are required to give your project a name and a description. If you choose to make your project discoverable by others, it is this information that they will be able to search, see, and use to get a sense of what it is that you are researching.

You can prevent the project from appearing in searches by selecting the **Exclude from public ORDS project list** box. If you select this, you will also be unable to expose any datasets you create within the project – if you want to be able to share data publicly, this box should be left unticked. You can, however, go back and change this setting (and all others) later.

The **Start date** and **End date** are for information only. Your project will not be deleted just because you have exceeded the given end date.

You should tick **Enable ODBC access for project** if you would like project members to be able to edit ORDS data using third-party software. This is discussed in a little more detail in section 6.

Once you have completed the form, click **Create**. The ORDS should return you to your project list, and you should be able to see your newly created project there.

In order to access a project from your list, simply click on its title.
The project page is where project administration is carried out. It consists of four main sections. At the top is the title and description. Then there is a list of databases in the project. When a project is first created, this will be empty. Next there is a list of the project members. Finally, there is a summary of the project metadata.

Some sections of the project page may not be visible unless your role on a project is owner or project administrator.

### 2.3.1. Trial projects and full projects

When you create a project in the ORDS system, it will initially have the status of a trial project. Trial projects have a number of restrictions. They are intended to allow you to get a feel for how the ORDS system works and whether it is suitable for use in your research.

In a trial project, you can:

- Import existing database files
- Create databases from scratch using the ORDS schema designer
- View, edit, and query data
- Work collaboratively by inviting other people to become members of your ORDS project

However, in a trial project, you cannot

- Add more than a hundred records to any database table
ORDS: An Introduction

- You can import database tables which contain over a hundred records to see how the data looks in the ORDS, but you will not be able to add any further records to these

- Make your data publicly available by creating and publishing ORDS datasets (sets of saved query results)

Additionally, trial projects are automatically excluded from the public ORDS project list.

If you decide that you would like to continue using the ORDS for your research, and wish to convert your trial project into a full project, click the button which appears towards the bottom of the project page.

There is no charge for upgrading trial projects, but you may be invited to come into IT Services for a brief chat about your project, to ensure that the ORDS is indeed an appropriate tool for your work.

Please note that tables in full projects are currently restricted to a maximum of 500,000 records. If your total data storage requirements are likely to be greater than 20GB (including saved datasets), please raise this as an issue when contacting ORDS staff.

2.4. Project members and their permissions

If you are the project owner or a project administrator, you can see a list of the project’s current members on the project page. Information is given about each member’s current role.

There are four roles that members of ORDS projects may have. Roles determine members’ permission levels – that is, what they can and cannot do within the project.

The **project owner** is the ORDS member with primary responsibility for a project. By default, the person who creates an ORDS project is its owner.

**Project administrators** have all the permissions that project owners have, except for the ability to delete the project.

**Contributors** can view, edit, and query data, create datasets, and export data, but they can’t alter the structure of a database nor edit the project or database metadata.

**Viewers** can view and query data, export data, and view datasets created by others, but they cannot edit any data themselves.

The permissions of the different roles are summarized in the table below:

<table>
<thead>
<tr>
<th></th>
<th>Project Owner</th>
<th>Project Administrator</th>
<th>Contributor</th>
<th>Viewer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete project</td>
<td>✔️</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>Add / remove / edit</td>
<td>✔️</td>
<td>✔️</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>project member</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create / delete</td>
<td>✔️</td>
<td>✔️</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>database</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create / delete</td>
<td>✔️</td>
<td>✔️</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>database versions</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Permissions are assigned at the project level. In other words, if someone is a project administrator, they will have project administrator permissions for every database within that project. There is at present no way to restrict permissions to a particular database.

### 2.4.1. Adding and removing project members

To add new members to a project you need to be the project owner or a project administrator.

To add a new member to a project, simply click on the `Add new member` button, then enter their email address and the role you would like them to have in the project (the role can be changed later). Then click on the `Add` button.

If the person with that email address is already registered with the ORDS, they will be added as a project member immediately. If they have not registered, they will be sent an email inviting them to register and join the project.

*Invited project members must register with the ORDS using the same email address as they were invited with, otherwise the system will not recognize that they should be added as a project member.*

To edit members’ roles, you also need to be the project owner or a project administrator.

Find the member whose role you wish to edit in the Project Members section of the project page. Click the `Edit project member` button that appears next to their name.

You will be taken to the Edit Project Member form. Select the desired new role from the drop-down list, and click on the `Save changes` button.

To delete a project member, simply click on the `Remove member` button next to their name. You will be asked to confirm this action before it is processed. Again, only project owners and project administrators can remove project members.

### 2.5. Project audit record

The ORDS system allows the project owner and project administrators to view the past audit record of a project. The audit record is a history of all the significant events that have happened within that project’s life – details of the project’s creation, adding and removal of members, creation and editing of databases, and so forth. It also allows you to see when each action occurred, and who performed it.
To view the audit record, find the **Project Details** section of the project page. Click the [View audit record] button.

After viewing the audit record, use your browser’s [Back] button to return to the project page.

### 2.6. Editing project details

If you wish to update the information you entered about your project during its creation, you can do so. Simply click [Edit project details], edit the fields as required, and then click [Save changes].

**Exercise 1: Registering to use the ORDS**

Now look at this exercise (page 59).

**Exercise 2: Creating a new project**

Now look at this exercise (page 61).
3 Creating databases

At the heart of the ORDS is the database. The ORDS enables you to import existing data in several common formats (see section 3.1 below for more details). Once you have imported a database or spreadsheet, you can then edit it, query it, and share it more widely.

You can also design databases from scratch using the ORDS’s database schema design tool.

However you wish to create your database, the first step is always to create a new empty database.

Whether or not you have already created any databases, you should always be able to see the **Add new ORDS database** button on every project page.

Click this to go to the **Add Database to Project** form.
As with the Create New Project form, the Add Database to Project form asks you for information that will help others find and understand your database in future. Even if you never intend to share the database, it is still good practice to record what it is, how the data was obtained, and where it came from – if for no other reason than to ensure that you do not forget this yourself a few years down the line.

Two fields in the form are mandatory: Name and Description.

The database Name can be whatever you choose, providing it is at least two characters long and does not include parentheses.

The database name and description will appear in the list of databases you and other project members see on the project page.

When you are happy with the contents of the form, click Add to create the database.

You will be taken back to the project page, and should see a message confirming that the database has successfully been created.

The name of the new database will now appear in the list of ORDS databases for the project. Note that because this is as yet an empty database, the number of database versions is currently zero.

Click on the database name to access the database page.

It should look something like this:
3.1. Importing an existing database

The ORDS allows you to upload existing databases either in Microsoft Access format or as PostgreSQL files. You can also upload one or more comma-separated value (.csv) files.

Pre-existing datasets in other formats will need to be converted to one of these three before they can be imported into the ORDS. It is usually easiest to do this by saving the data as .csv files – Microsoft Excel spreadsheets, for example, can be converted to .csv files by using the Save As function and selecting CSV (Comma delimited) as the file type.

To begin uploading a database, click Upload existing database. You will see the Upload a Database to ORDS page.
Click ![Browse...](#) to select the file you would like to upload (the labelling of this button may vary depending on the browser you are using). Then click ![Upload](#) to begin the actual upload process.

When you upload a database into ORDS, it is automatically converted to PostgreSQL format. This can take a few minutes, so the system will email you once the upload and conversion are complete.

PostgreSQL is stricter than programmes such as Microsoft Access in terms of what it regards as a valid database. The ORDS checks for some of the common problems encountered when recreating an Access database as a PostgreSQL database, but it can't catch everything, so you might sometimes receive an email saying that the upload process has failed. If this happens, the ORDS support team might be able to fix the original database to enable it to upload smoothly. Feel free to contact the team at ords@it.ox.ac.uk.

If you click the database name to return to the database page before the upload process has completed, you will see a message confirming that the database is still in the process of being uploaded. Once the upload has finished, the database page will look more like this:

![Database Name: New database](#)

Note that the buttons under the **Main database version information** heading have changed, and there are now sections for a **Milestone database version** and a **Test database version**.

### 3.2. Importing data in spreadsheets

Before you try to upload a .csv file, you need to ensure that it is in a form that the ORDS can understand.
If you created your data in *Microsoft Excel* or other commonly-used spreadsheet software, you may need to delete any extraneous information from your spreadsheet before exporting it as .csv. The first row of your spreadsheet should contain the field (or column) names, listed horizontally from column 1 to n. Your data should be entered in the rows numbered 2 to n, immediately below the column headers. You should not leave any blank rows or columns. Empty cells within your data grid are fine – they will be interpreted as null values when uploaded.

Remove from your spreadsheet any additional information such as titles, notes, calculations off to one side of the main table, and that sort of thing. Only once you have a ‘pure’ table remaining should you save your data as a comma-separated value (.csv) file. Do make sure you’ve saved a copy of any important contextual information somewhere else – if you choose, you can add this as comments on tables or individual fields once the data is in the *ORDS*.

As with a database, to begin uploading spreadsheets, click [Upload existing database](#). You will see the **Upload a Database to ORDS** page.

Click [Browse](#) to select the .csv file you would like to upload, and then [Upload](#) to begin the actual upload process.

As before, the system will email you once the upload and conversion has completed.

Once the upload process has finished, a new option will appear on the database page. You can now upload additional spreadsheets by clicking [Upload another CSV file as a new table](#).

Only one spreadsheet can be uploaded at a time, but you can repeat the process as many times as necessary to upload multiple spreadsheets into a single database. Each spreadsheet will become a separate table within the database. You can later create relationships between the separate tables to turn your spreadsheets into a proper relational database, if you so choose.
Be aware that you cannot upload both an Access or SQL file and .csv files into the same ORDS database. If you need to do this, ask the ORDS team for assistance (ords@it.ox.ac.uk).

3.3. Using the database schema editor

To access the database schema editor, click **Create new database** if you wish to create a new database from scratch, or **View and edit database structure** if you have already uploaded a pre-existing database. You will then be taken to the database schema designer, which will look a bit like this:
If you are working with an existing database, the structure of the database will be uploaded into the working area by default. If you are creating a new database from scratch, it will be empty, as in the illustration above.

Using the schema editor, you can add and delete tables, add and remove fields within tables, change field names and types, and express relationships between tables. Remember to [Save] any changes you make, otherwise they will be lost when you leave the designer and your database will revert to the form it was previously in.

If you try to make any changes to your database structure that are not permitted by the rules of PostgreSQL, you will receive an error message. The system will also complain if you try to change a field type to something which conflicts with data already in that field in the database – so, for instance, if you try to make a field into one that can only accept whole numbers (known as an Integer field), but some of the data in that field is textual, you will receive an error message.

**IF YOU REMOVE A FIELD OR TABLE AND THEN SAVE THE NEW SCHEMA, ANY DATA THAT WAS IN THAT FIELD OR TABLE WILL BE DELETED.**

You will be warned before this happens, but once the data is gone, it is gone. Clicking CTRL+Z to try to undo the deletion will not work. It is therefore a good idea to create a test version of your database before doing any re-design work that might result in data loss; that way, if you make a mistake, this will not affect the main version of the database. For more about creating and working with versions of database, see section 3.4.

### 3.3.1. Adding and removing tables

To add a new table, click the **Add table** button on the toolbar. The mouse pointer will change from an arrow to a cross.
Click the place on the working area where you would like the table to appear. You will be able to move it later if you wish.

The new table will appear, and the **Edit table** dialogue box will open.

![Edit table dialogue box]

You will see that the table has been given a placeholder name – `new_table` followed by a number. Replace this with the name you would like the table to have.

The **ORDS** system supports the use of spaces in table names, but this is not true of all database systems. If you are likely to want to export the data for use in other systems later on, it may be worth bearing this in mind.

It is also usually best to avoid punctuation marks or other special characters in table names, as these can sometimes cause problems. One exception to this is the underscore: these can be used safely in most systems, and are often used as a replacement for spaces.

Some people like to follow the convention of beginning every table name with the letters `tbl` – this can be done or not, according to personal preference.

You also have the option of adding a comment about the database table. This can be used to record metadata or notes about the table contents. Comments can be added or edited later.

Click **OK** to close the dialogue box. You will see that your table has been automatically populated with a field called `id`.

![Table with id field]

This will be the primary key field for the table – the field that provides a unique identifier for each record. The name of the primary key in an **ORDS** table is shown in bold type. You can change the field name if you wish.

To view a comment on a table, hover the mouse pointer over the table header. (You can also click the appropriate field to select it, and then click `Edit table`).

![Comment on a table]

To delete a table, click the table you wish to delete (so that it is highlighted) and then click **Remove table**.
3.3.2. Adding new fields

To add a new field to a database table, select the table you wish to add a field to by clicking its header. The border of the selected table will darken, and the rectangle representing the table in the thumbnail map will turn orange.

Now click the `Add field` button in the toolbar. (If no table is selected, the button will be greyed out.)

The table will expand to show the options available when adding a new field.

The field will have the placeholder name of `new_field`. Replace this with a name of your choice. As with table names, it is generally best to avoid special characters (other than underscores) in field names, and you may wish to avoid spaces if you are likely to want to export the data for use elsewhere.

The remaining fields in the dialogue box allow you to select a data type, and to set a number of other options.

3.3.3. Data types

Like most database software, the ORDS expects each field within each database table to be of a defined type. Some fields, for instance may be defined such that they will only accept numerical data. Others may be text fields, into which users can enter any text (including numbers). The type of field you define will affect how people can search your data. If you define a field as numeric, people will be able to search it by criteria such as only showing results where the data in that field is greater than a hundred; if you enter numbers into a text field, searches such as this will not work, because the system will treat the number as though it is a string of text.

By choosing an appropriate data type for your data, you can prevent people from adding inappropriate data to your database. For example, you may wish to specify that a ‘date of birth’ field can only contain a date.

The full list of ORDS data types is given below, although in practice many databases will only use a handful of different data types – Integer and Text being the most common.

<table>
<thead>
<tr>
<th>ORDs basic data types</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Integer</strong></td>
<td>A whole number, negative or positive. This data type can be used for numbers between -2,147,483,648 and 2,147,483,647.</td>
</tr>
<tr>
<td>Data Type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Decimal</strong></td>
<td>A number (negative or positive) which may include a decimal point. This field type can store numbers with up to 13,107,2 digits before the decimal point, and up to 16,383 digits after it, so is recommended for cases where a high degree of accuracy is required. It is possible to specify a maximum total number of digits in the number (including those before and after the decimal point), and a maximum number of digits after the decimal point. This is done by entering two figures, separated by a comma, in the Size field in the ORDS schema designer. In some cases, it may be desirable to restrict the number of digits after the decimal point, but not before. This effect can be achieved by using an arbitrarily large number – e.g. 100 or 1000 – before the comma, and the required number of decimal places after it. So, for example, for a field designed to display an amount of money given in pounds and pence, one might enter ‘1000,2’ in the Size field. This would allow up to 1000 digits in total, but only 2 of those after the decimal point.</td>
</tr>
<tr>
<td><strong>Text</strong></td>
<td>Fields with this data type can contain text, numbers, or a combination of these. This is the most flexible data type, so is often the best option if you are uncertain exactly what a field may need to contain. The length of Text fields is not restricted.</td>
</tr>
</tbody>
</table>

**All ORDS numeric data types (coloured yellow)**

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Integer</strong></td>
<td>A whole number, negative or positive. This data type can be used for numbers between -2,147,483,648 and 2,147,483,647.</td>
</tr>
<tr>
<td><strong>Big integer</strong></td>
<td>A whole number, negative or positive. This data type can be used for numbers between -9,223,372,036,854,775,808 and 9,223,372,036,854,775,807. Use when the range permitted for the integer data type is not sufficient.</td>
</tr>
<tr>
<td><strong>Decimal</strong></td>
<td>A number (negative or positive) which may include a decimal point. This field type can store numbers with up to 13,107,2 digits before the decimal point, and up to 16,383 digits after it, so is recommended for cases where a high degree of accuracy is required. It is possible to specify a maximum total number of digits in the number (including those before and after the decimal point), and a maximum number of digits after the decimal point. This is done by entering two figures, separated by a comma, in the Size field in the ORDS schema designer. In some cases, it may be desirable to restrict the number of digits after the decimal point, but not before. This effect can be achieved by using an arbitrarily large number – e.g. 100 or 1000 – before the comma, and the required number of decimal places after it. So, for example, for a field designed to display an amount of money given in pounds and pence, one might enter ‘1000,2’ in the Size field. This would allow up to 1000 digits in total, but only 2 of those after the decimal point.</td>
</tr>
<tr>
<td><strong>Single precision</strong></td>
<td>A number (negative or positive) which may include a decimal point. Although this type of field (known as a floating point type) can be used to store very large or very small numbers, the precision (i.e. the number of significant figures, before and after the decimal point) is...</td>
</tr>
</tbody>
</table>
### Double precision

A number (negative or positive) which may include a decimal point.

Although this type of field (known as a floating point type) can be used to store very large or very small numbers, the precision (i.e. the number of significant figures, before and after the decimal point) is limited to 15 digits. In practice, this means that rounding of large or small numbers may occur.

This data type is included chiefly for compatibility with other database systems. When creating a new field, it will generally be preferable to use the data type **Decimal**.

### All ORDS character (text) data types (coloured pink)

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Char</strong></td>
<td>Also known as <strong>Character</strong>, fields of this type can contain text, numbers, or a combination of these. <strong>Char</strong> fields have a fixed length, set by specifying a number of characters in the <strong>Size</strong> field in the ORDS schema designer. If the entry in a <strong>Char</strong> field is longer than the specified number of characters, it will cause an error; if it is shorter, the field will be padded out with spaces. If no field length is specified, the default is for the field to be limited to a single character. This data type is included chiefly for compatibility with other database systems. In some other systems, using the <strong>Char</strong> field type confers performance advantages. However, this is not true for PostgreSQL systems such as the ORDS (in fact, the padding will increase the file size), and hence for ORDS databases it will generally be better to use the <strong>Varchar</strong> or <strong>Text</strong> field types.</td>
</tr>
<tr>
<td><strong>Varchar</strong></td>
<td>Also known as <strong>Character varying</strong>, fields of this type can contain text, numbers, or a combination of these. It is possible (though not required) to set a maximum length for a <strong>Varchar</strong> field, by specifying a number of characters in the <strong>Size</strong> field in the ORDS schema designer. Entries which are longer than the specified number of characters will be truncated. In PostgreSQL database systems such as the ORDS, there is no practical difference between a <strong>Varchar</strong> field with no length limit and a <strong>Text</strong> field. In some other systems, however, these are handled differently, and hence both options are included to increase compatibility with other systems.</td>
</tr>
<tr>
<td><strong>Text</strong></td>
<td>Fields with this data type can contain text, numbers, or a combination of these. This is the most flexible data type, so is often the best option if you are uncertain exactly what a field may need to contain. The length of <strong>Text</strong> fields is not restricted.</td>
</tr>
</tbody>
</table>

### All ORDS date and time data types (coloured green)

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Date</strong></td>
<td>A date consisting of a day, month, and year. Dates should be entered in the YYYY-MM-DD format.</td>
</tr>
</tbody>
</table>
If you wish to display dates in other formats, use the **Text** data type.

<table>
<thead>
<tr>
<th>Time</th>
<th>A time of day, consisting of hours, minutes, and seconds. Times should be entered in the format hh:mm:ss. If you wish to display times in other formats, use the <strong>Text</strong> data type.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Timestamp</strong></td>
<td>A concatenation of a date and a time. Timestamps should be entered in the format YYYY-MM-DD hh:mm:ss.</td>
</tr>
</tbody>
</table>

**ORDS miscellaneous data types (coloured purple)**

<table>
<thead>
<tr>
<th><strong>Boolean</strong></th>
<th>Data entered in fields of this type is restricted to one of a pair of values. Depending on the context, the pair may be interpreted as true or false, yes or no, or 0 or 1. In an ORDS database table, a field with the <strong>Boolean</strong> data type will be displayed as a check box.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Binary data</strong></td>
<td>Used for raw binary data (that is, data consisting of os and 1s), stored as hexadecimal strings. Images and audio files are examples of binary data.</td>
</tr>
</tbody>
</table>

It is worth noting that there are certain types of data which may at first sight appear to be numerical, but for which a field with the **Text** data type is actually more suitable. These include:

- Phone numbers. These generally begin with a zero, which will be ignored if the field has a numeric data type. They may also include other non-numeric characters, such as spaces or brackets.
- Number ranges – for example, page ranges in citations, or date ranges. These will include a dash or other non-numeric separator.
- Information which is chiefly numeric, but which may sometimes need to be qualified. For example, a ‘Year’ field might normally contain a single four-digit number. Occasionally, however, the precise date might be uncertain, and you may wish to indicate this by putting ‘ca.’ before the number.

The schema designer can be set to display the data type of each field. To turn on this feature, click the **Options** button in the toolbar, and then check the **Show field data type in table design?** checkbox. Clicking this checkbox again will clear the tick and turn this feature off. You will need to refresh your browser to see the data type labels (usually by pressing F5 on the keyboard).

### 3.3.4. Other ORDS field options

**Size:** Some ORDS data types also allow you to set the size of the field. This is done by entering a number (or for certain field types, two numbers separated by a comma) in the **Size** box, to specify a maximum number of characters. If it is not possible to specify a size, this field will be greyed out.

The schema designer can be set to display the size of each field for which a size has been set. To turn on this feature, click the **Options** button on the toolbar, and then click the **Show field size in table design?** checkbox so that a tick appears in it. Clicking this checkbox again will clear the tick and turn this feature off.
feature off. Click [OK] to close the dialogue box. You will need to refresh your browser to see the size labels.

**Default:** This option allows you to specify what will appear in this field if no other data is added. When you first create a field, this will be set to **NULL**, indicating that the field is empty. You can change this to something else if you wish (for example, you might wish the default entry to be 'Not known').

**Autoincrement:** This option is only available for **Integer** fields. If this option is selected, then when a new record is added, the **ORDS** will automatically complete this field with the next available number.

*When you create a new table, the id field is set by default to be an autoincrement Integer field. This is the most straightforward way of ensuring each record has a unique primary key value.*

**Permit null values:** In a database, a field with a null value is one that is currently empty. (This should be distinguished from 0, which indicates that the field has been completed with a value of zero.) If this checkbox is selected, null values will be permitted in this field. If it is deselected, the field will be mandatory – the **ORDS** will not allow a record to be saved unless the field has been completed.

**Comment:** You can add a comment to an individual field. This can be used to store notes about the field or the data in it – for example, you might use it to explain an abbreviation in the field name, or to indicate a unit of measurement.

To add or edit a comment, click [Edit comment], type your desired content in the dialogue box which appears, and then click [OK]. The comment will appear at the bottom of the field options dialogue box.

To view a comment again later, hover the mouse pointer over the field. (You can also click the appropriate field to select it, and then click [Edit field].) Comments are also visible in the data editing interface.

Once you are happy with your field name and choice of options, click [Done]. Your new field will be shown at the bottom of the table.
3.3.5. Relationships between tables

Relationships lie at the heart of a relational database. Whilst you can use the ORDS simply as a way of hosting and querying multiple flat files (essentially spreadsheets), it is when you combine tables into a relational database that you can unlock the unique advantages of this more sophisticated structure.

Relationships are a way of linking database tables. Two types of relationships are commonly used: one-to-many, and many-to-many.

One-to-many relationships exist between a single entity of one kind, and multiple entities of another. For example, a child can only have one biological mother, but a mother may have many children. (Some mothers, of course, may only have one child, but this does not change the nature of the relationship – the important point is that it is possible for a mother to have multiple children.)

Many-to-many relationships exist between multiple entities of one kind and multiple entities of another. For example, a brother may have multiple sisters, and a sister may have multiple brothers. (Again, the fact that some brothers may have only one sister – or none at all – does not affect the nature of the relationship.)

It is also possible to have a one-to-one relationship within a database. In practice, however, these are rarely used. Within the ORDS system, a one-to-one relationship would be implemented in the same way as a one-to-many relationship, but no entity would be linked to more than one entity of the other type.

By expressing relationships between different types of entities, you can avoid typing in the same data into many different cells and avoid the scope for inconsistencies that this creates. Defined relationships also enable far more flexible and interesting ways of querying your data.

**Adding one-to-many relationship in an ORDS database**

Suppose that a database includes a table which has a number of fields for details of books in a collection. (The table shown below has the **Show field data type in table design?** option turned on – see section 3.3.3 above for details.)
Consider the Publisher field. Each book will have a single publisher, but each publisher is likely to have published many books. This is therefore a one-to-many relationship.

The first step is to create a second table, with fields for the information we would like to record about the publishers.

We now need to join the two together. It is best to think of a one-to-many relationship as linking the entries in one field in one table (in this case, the Publisher field in tblbooks) with whole records from another table (in this case, tblpublisher).

We will therefore create the link between the Publisher field in tblbooks and the id field in tblpublishers. The id field is used because as the primary key for the tblpublishers, this is what the ORDS uses to uniquely identify the records in that table. (The contents of the primary key field can be thought of as representing the record as a whole.)

A field that is linked to the primary key of another table is known as a foreign key. So in this case, the foreign key is the Publisher field.

When creating the relationship, it will be helpful to have the two tables side by side in the schema designer. Tables can be moved by clicking the header, holding the mouse button down, and dragging the table to the chosen location.

The next step is to ensure that the two fields to be linked have the same data type. This means we need to change the data type of the Publisher field to Integer. (If this seems counter-intuitive, remember that the tblpublishers id field represents the whole record: it is an Integer field representing a number of fields of various types.)

To do this, click the Publisher field to select it (two small red arrows will appear at the left-hand end of the field), and then click the Edit field button in the toolbar. Use the Type drop-down list to change the data type to Integer, then click Done.
Now click the id field of tblpublishers to select it. This will activate the Connect foreign key button on the toolbar; click this button.

To create the relationship, click the Publisher field in tblbooks. A curved line will appear, joining the two fields.

It is always a good idea to Save your database structure after establishing a relationship.

**Adding many-to-many relationships**

As in most database management systems, many-to-many relationships are handled in the ORDS by creating a link table.

Suppose that we wanted to add a relationship between tblbooks and another table listing author details. First, let’s create the table for the author information.
(For convenience, this table has been placed to the left of `tblbooks`. The arrangement of the tables in the schema designer working area does not change the database structure; tables can be moved around as desired.)

If a relationship was created between the `id` field in `tblauthors` and the `Author` field in `tblbooks`, this would be a one-to-many relationship: it would allow the possibility of each author having written multiple books, but it would only be possible to list a single author for each book. However, some books have multiple authors – this is therefore a many-to-many relationship.

To represent this many-to-many relationship, another table is needed. Records in this table will store combinations of authors and books. For example, if *A Natural History of Great Britain* was written by Smith, Jones, and Brown, three records would be added to this table, one for each author-book combination.

It will be easiest to add the necessary relationships if the new table is positioned between `tblauthors` and `tblbooks`.

We now need to add two further fields to the new table: one which will be joined to the `id` field of `tblauthors`, and one which will be joined to the `id` field of `tblbooks`.

The quickest way to do this is to create new foreign keys in `tblbookauthors`.

To create a foreign key for the `tblauthors id` field, click `tblauthors id`, then [Create foreign key], then the header of `tblbookauthors`. The foreign key field will appear in `tblbookauthors`, with a link to the `tblauthors id` field.
Repeat this process for the `tblbooks id` field.

Link tables will often consist only of their own primary key field, plus two foreign key fields. However, it can sometimes be useful to add additional fields. For example, we might add a `Role` field to `tblbookauthors`, which could be used to indicate the part someone had played in the writing of a book (main author, co-author, editor, translator, etc.).

The `tblbookauthors` table would be the most appropriate place for this field, as being (for example) the main author is a property of a particular act of authoring, rather than of an author — because an author might be the main author of one book, and the co-author of another.
Information about the authors of various books will be retrieved by running a database query. The Author field in tblbooks is now superfluous, so needs to be deleted to avoid confusion.

As this a new (and therefore empty) database, this can be done immediately. However, if this were an existing database which already contained records, it would be important first to make sure a copy of all the relevant information had been saved, so it could be used to populate the new tables.

To delete the Author field, click it to select it, then click the button on the toolbar. You will be asked to confirm that you really do wish to delete the field. To continue, click OK.

3.3.6. Saving changes to a database structure

The schema designer does not automatically save changes. It is therefore important to ensure you save the schema before leaving the page, or any edits you have made will be lost.

To save your schema, click the button on the toolbar. You will see a dialogue box asking you to confirm that you wish to overwrite your existing database structure with the updated version (you will see this even if the database is brand new, and there is nothing to overwrite).

Click OK.

ORDS will then attempt to save the database.

If the save operation is successful, you will see a message saying Changes saved. Click OK to close the dialogue box.

If there is a problem with the database structure, the save operation will fail, and you will see an error message. You will need to find and fix any problems before you can save the schema.

It is a good idea to get into the habit of saving the schema regularly as you design your database. This reduces the risk of losing work in the event of a problem with the database structure or a system error.

If you cannot resolve the problems with your database structure, or think you have encountered an error which is preventing you from saving changes, please ask the course instructor for help, or (if you are using the system outside an ITLP course) contact the ORDS help desk by emailing ords@it.ox.ac.uk.

If you leave the schema designer open for an extended period without working on it, there is a risk that the system will time out, and you will find you are unable to save any changes previously made. It is therefore sensible to save changes whenever you step away from your computer, or if you change windows to work on another task.
3.3.7. **Common problems when changing database structures**

If you find that the system will not save your database structure, this is likely to be because:

- You have tried to change a variable type but there is already data in that field which is incompatible with the new data type.
- You have two tables with the same name, or two fields with the same name within one table.
- You are trying to create a relationship between two fields that are not of the same variable type.
- Your browser session has timed out – this may happen if you’ve been working on the structure for a long period without saving changes. Do remember to save changes frequently – this also makes it easier to see what might be going wrong if a problem does occur.
- You have more data in a particular table than the ORDS can hold in memory (usually around 500,000 records).

If after checking your structure none of these problems are evident, ask the course instructor for help, or write to ords@it.ox.ac.uk explaining the problem. (Including a screenshot of your structure might help the ORDS team diagnose the problem. On most computers you can take a screenshot by pressing the PRNTSCRN function key and then pasting the image into a word processor file using CTRL+V.)

3.4. **Milestone and test versions of databases**

If you are a project owner or administrator, you can create and delete database versions, and set a test database version to be the main database version.

The ORDS system allows you to have up to three versions of each database. When you upload a database or create one using the schema designer, this automatically becomes the main database version. You then also have the option of creating a milestone version, and/or a test version.

At the point when a milestone or test version is created, it will be an exact copy of the main database version at that time. However, after that point, the versions then operate independently: any changes made to the contents or structure of the main version will not affect test or milestone versions, and vice versa.

**Main database version**

As the name suggests, this is the primary copy of the database. Generally, most of the work (adding or editing records, querying, creating datasets, etc.) will be done with this version.

**Milestone database version**

A milestone database version is intended to provide a snapshot of the database as it was at a particular point. This can be used to enable comparison of past and present versions of the data.

**Test database version**

A test database version is intended to provide a safe environment to experiment with changes before implementing these in the main version. For example, if a major change to the structure of the database is planned, it is best to create a test version and to try the changes here first. If the changes are successful, the test
version can be set to be the main database version. Alternatively, the test version can simply be deleted, leaving the main version intact and unchanged.

3.4.1. Creating and deleting a milestone or test version of a database

To create a milestone version or test version of a main database version, simply click on the relevant button in the database page:

Clicking on either button should result in a ‘success’ message, and you will see additional buttons appear in the Milestone or Test panes. These work in the same way as the corresponding buttons in the Main database pane, but will only apply to the database version under which they are listed.

If you create a test version of your database you will see one additional button: Clicking on this button will make the current test version of the database into the new main version, overwriting the design and all of the data currently in the main version. You will be asked to confirm that this is really what you want to do.

Clicking the button in the Milestone or Test pane will delete that version and all of the data therein. Clicking the button in the Main database version pane will delete ALL versions of the database and all of the data. Only do this if you are sure that you want to start this database again from scratch.

3.5. Exporting databases

If you wish to load your database into another database management system or simply keep a local copy on your own machine, you can export it in its entirety. You can only export the main version of a database, not the milestone or test versions.

Simply click the button under the Main database version pane and select the file format you would like to use for your export.
There are three different database export formats:

- Multiple .csv files
- A single .csv file
- A PostgreSQL dump

.csv files are simple comma-separated value files. There is a huge variety of software that will allow a .csv upload, including database software, spreadsheet software, data analysis software, and so forth. The only problem with .csv files is that they will not retain information about the structure of your database - just the table names, field names, and the data itself. Any relationships you have established between tables will be lost.

Downloading your database as a single .csv file will combine all tables in your database into a single file. Downloading multiple .csv files will create a zipped download containing a separate file for each table.

A PostgreSQL dump exports all of your data along with information about the structure of your data, including the relationships between tables. This is the ideal choice if you want to retain all information about your database. The file will be saved with the extension .sql. Because PostgreSQL is an open standard, you should find that there is a wide range of software available into which you can upload your database file.

Select your desired export format, and click **Export** to save your database to your computer.

On a *Windows* machine, the file will be downloaded to your *Downloads* folder by default.

**Exercise 3: Importing a database file**

Now look at this exercise (page 62).

**Exercise 4: Viewing and editing the database structure**

Now look at this exercise (page 64).
4 Editing data

Once you are happy with your database structure, you probably won’t go back to use the schema designer very frequently. Most of your work will be done using the data editing interface.

4.1. The ORDS editing screen

To view and edit data in the main database version, click the View, edit, and query data button in the main database panel on the database page.

You will be taken to the View, Edit, and Query Data page, where you will see a list of the tables in the database.

To view the contents of a table, click the table name. The data contained in the table will be displayed.
If the database table currently contains no records, you will see just the table description and the **Add new record** button.

If, on the other hand, the table contains more than a hundred records, these will be divided into pages. At the top of the data table display window, you will see a summary of which records you are currently viewing, the total number of records in the data table.

You can move between pages using the **Previous** and **Next** buttons. You can also skip to a specific record by typing the record number into the **Go to Record** box and then pressing **ENTER**.

Beneath the pagination bar, you will see the actual data records in the selected table. To edit the records on-screen, simply click in the cell you wish to edit and then change the data to the desired value. The cell will turn blue to indicate that the data has been altered. You can edit more than one cell at a time before saving your changes.

To save your edits, press **ENTER**, or click on the **Save changes** button. The table will re-load and you should see your changes in the display.

Note that if you try to enter data in a cell that is not compatible with the data type for that field, the **ORDS** will give you an error message and your edits will not be saved.
4.2. Editing data in linked tables

If you are editing data in a table that has relationships with other tables in your database, you will notice that the linked fields work slightly differently from unlinked fields.

The database table shown below is a list of articles from a bibliographic database. Each record includes information about the article’s author, and the journal in which the article was published. This information takes the form of a link to another table in the database – **tblauthors** for the author information, and **tbljournals** for the journal information.

By default, the **author** and **journal** fields for each record in the **tblarticles** table display the contents of the primary key fields of the **tblauthors** and **tbljournals** records. In this case, this simply contains an ID number - which is not very informative if you are trying to edit the records.

To change this, you can click the drop-down list at the top of the column, and select the field from the linked table that you would like to view.
The table should now update to display the data from the selected field.

Where a field is linked to another table, you cannot simply write into that field the value you would like: you must select it from the records in the linked field. This ensures the integrity of the database, and it is also a useful way to ensure that data editors are consistent – there is no opportunity for typos or alternative forms of words or names. To select a value, click the small arrow at the right-hand end of the field: the ORDS will display a drop-down list of the available options.

If you have a lot of records to select from in the linked table, then as well as a drop-down list, the ORDS will also provide a search box. You can start typing the contents of the record you are looking for into this: this will narrow down the available options.
If you wish to add a new option to select, you will need to add that as a new record in the source table.

4.3. Filtering records

Beneath the table, at the bottom of the screen, you will see the record filtering interface. This allows you to select and view a subset of records. This can be particularly useful if you are working with large datasets.

The default filter interface consists of two drop-down lists and a free text field. The first drop-down list allows you to select the field on which you wish you filter the records. The second drop-down list contains a list of operators. These will vary depending on the data type of the field selected, but include options such as is (for exact matches), is not, contains, greater than, less than, and is NULL. The free-text field is where the search criteria are entered.

For example, to find all articles in tblarticles whose title includes the word 'Church', you would select title from the first drop-down list, and contains from the second, then type Church in the free-text field. (The filter is case sensitive, so Church and church will return different results.)

To apply the filter, click Apply filter.
ORDS: An Introduction

This filter reduces the number of records displayed from fifteen to three.

To return to viewing all records in the table, click the Clear filter button.

It is possible to apply multiple search criteria simultaneously. To add an extra row to the filter interface, click the blue plus symbol at the right-hand end.

In addition to the extra row, a third drop-down menu will also appear, allowing you to specify whether you wish to link the two criteria using an AND or an OR.

Additional search criteria can be added by clicking a plus symbol again, allowing complex Boolean searches to be constructed. For example, the filter below will find records for all articles published in the 1990s which have either Britain or England in the title.
Criteria can be removed by clicking the minus symbol.

It is also possible to filter records using the content of linked fields, but this requires a slightly different approach.

The field selection drop-down list does contain entries for linked fields. For this table, which has two linked fields, we can see that both of them - author and journal - appear in the list.

However, these can only be used for filtering on the contents of the primary key field of the linked-to table. To filter on the content of other fields from the linked-to table, we need to first select the table, and then the desired field. At the top of the field selection drop-down list, you will see a list of the other tables that are linked to from this one - in this case, tbljournals and tblauthors. To distinguish table names from ordinary field names, each table name is followed by an ellipsis.
If we select one of the tables, an extra drop-down list is added to the interface.

This contains a list of all fields from the linked-to table, allowing us to select the one we'd like to filter on. We'll select the author_name field.

We can now select an operator, and add filter criteria to the final box - just as we would when applying any other filter.
We can now apply the filter, and see the results. This filter shows us all records in `tblarticles` whose `author_name` field contains *French*. (After applying a filter, you may sometimes find you need to reselect the fields from linked-to tables that you wish to view - see above for details.)

Filters on linked fields can be included in combined filters, as described above. Note: if a table contains two or more linked fields which link to the same table, you may find filtering on linked fields does not work as expected. We are aware of this issue, and are working to fix it as soon as possible.
4.4. Adding and deleting records

If you are the project owner, a project administrator, or a contributor, you can add new records to database tables.

Open the table to which you wish to add a new record. Click the button, which is located towards the bottom of the window. This will take you to the Add New Record form.

This form is generated automatically according to the types of field in the table. For each field you will see the field name with the type of data it takes indicated below.

Fields that are linked to other tables include a drop-down list of the records you can select from, with another drop-down list below and to the left of the field, which you can use to select the column from the linked-to table that you wish to view the contents of.

Fields with certain special data types, such as Date, Time, or Timestamp, indicate the format in which you should enter the data in light grey text (e.g. hh:mm:ss for Time), along with a time- or date-picker tool to the right.

Auto-increment fields are greyed-out: the ORDS will automatically assign a unique number to such fields when you add the record. This prevents two records being created with the same number when you have multiple project members adding data at the same time. If you wish to be able to input a number of your own choosing in such fields, you will need to go to the schema designer and deselect the auto-increment box.

Boolean data fields only accept True or False values. A check box is provided.

Once you are happy with the contents of the form, save the new record by clicking either Save and add another or Save and return to table, as desired.
To delete a record, go back to the data editing interface, and click the Delete button at the right-hand end of the record you wish to delete. You will be asked to confirm the deletion before proceeding.

If a record is linked to by records in other tables, ORDS will not let you delete it: after clicking, you will instead see an error message. To be able to delete the record, you will first need to delete or edit all the other records that link to it.

4.5. Adding new records in bulk

At present, each new record you add to a table needs to be added individually. If you wish to add or upload records in bulk, contact the ORDS team at ords@it.ox.ac.uk for support. They may be able to write a script that will add a large number of new records to an existing table.

Exercise 5: Viewing and editing data

Now look at this exercise (page 67).
5 Querying a database

All ORDS project members are permitted to query databases within the project.

From the database page, click the green button under the database version you wish to query.

You will be taken to the View, Edit, and Query Data page, where you will see a list of the tables in the database, plus a number of buttons. Two of these buttons -- Query data using query builder and Query data using SQL -- will take you to ORDS query tools.

The Query Builder allows you to construct simple queries using drop-down lists. It is straightforward to use, but its scope is limited. The SQL Query Page is much more powerful, but requires knowledge of SQL (Structured Query Language). The Query Builder will be further developed during 2015-6 to enable the construction of a wider range of query types.

5.1. The Query Builder

The Query Builder tool enables the construction of SQL queries without the need to learn SQL. It allows you to select one or more tables from your database, and view the contents of some or all of the fields.

The query builder has three main components:

The working area occupies the top left-hand part of the screen. The database tables will appear here. If you need more space, you can increase the size of the working area by dragging its lower right-hand corner.

The table list normally appears in the top right-hand part of the screen (in narrow windows, it may move to below the working area). You will see a list of all the tables in the database.

The query box appears towards the bottom of the screen. As you select fields to be included in the query, you will see an SQL statement appear here.
To include a table in the query, click the check box that appears next to the table’s name in the table list. To remove a table from the query, click the check box again to uncheck it.

Let’s begin with a simple query which only uses a single table - `tblarticles`. Once we have checked the `tblarticles` box in the table list, the table appears in the working area.

The table contains a list of all its fields, each with its own check box. We’ll check all of them, to include the whole contents of the table in this query.
The query builder has generated an SQL statement reflecting the selection of fields. This can be seen in the query box. We can now click [Run query] to see the results.

The query results will appear in a new tab.

All the fields are present, but notice that the journal and author fields only contain numbers. This is because these are both linked fields, which draw on information from other tables. The journal and author fields are displaying the contents of the primary key field of the linked-to records, which contains only a number (like tblarticles's own id field, which can also be seen in the table.
above), and hence isn’t very informative. Fortunately, we can use the query builder to construct a query that will give us the contents of other fields from the linked-to tables.

Let’s return to the query builder and select some additional tables.

The relationships between the tables are represented by curved lines. (tblarticles is now on the right. Tables can be dragged around the working area as desired - it is often helpful to do this to arrange the tables so the relationships can be seen clearly.) We can see from the diagram that the journal field in tblarticles links to tbljournals, and the author field to tblauthors.

We now need to select the additional fields we’d like to appear in our query. We’ll select the journal_name and publisher fields from tbljournals, and the author_name field from tblauthors. (We could also select any of the others that we’d like to include.) Additionally, we’ll deselect the journal and author fields from tblarticles, as we already know that these only contain an uninformative primary key.
The SQL statement in the query box has updated to reflect this new, more complex query. We can now click again, to refresh the contents of the results tab.

These results are rather more informative!

We can sort the results by one of the fields by clicking on the field name - a small arrow will appear next to it. Clicking the same field name a second time reverses the direction of the sort (and also inverts the arrow). Let’s sort the results by the author_name field.

Although the query builder is currently somewhat limited in scope, it is also possible to use it as a starting point for creating more complex queries.

The SQL statement generated by the query builder can be edited manually. For example, it is possible to change the order in which fields appear in the query results by editing the order in which the table and field names appear in the
statement (ensuring that the punctuation around these remains intact). Those with further knowledge of SQL can also add additional clauses - e.g. **WHERE** or **ORDER BY** - to the statement before running the query.

If you need assistance with querying your database, the ORDS team may be able to help: please contact the help desk by emailing ords@it.ox.ac.uk.

### 5.2. SQL queries

The **SQL Query Page** is a flexible tool, which allows more complex queries to be constructed. These include queries which draw data from multiple tables, or which include conditional statements.

Although SQL (Structured Query Language) may initially look slightly cryptic, it is possible to gain a reasonable grasp of the main principles relatively quickly. Plenty of books and online guides to the language are available. In particular, W3schools.com offers a helpful tutorial at [http://www.w3schools.com/sql/](http://www.w3schools.com/sql/).

If you need assistance with querying your database, the ORDS team may be able to help: please contact the help desk by emailing ords@it.ox.ac.uk.

To open the **SQL Query Page**, click **Query data using SQL** from the **View, Edit, and Query Data page**.

The page provides separate boxes for each statement of an SQL query, with some help text (and links to further information) on the right-hand size.

When typing in an SQL query, there is no need to enter the SQL command that appears immediately before the box (e.g. **SELECT**, **FROM**, etc.): the ORDS will supply this automatically. If you do enter one of these commands, the ORDS will delete it.
If the ORDS detects that you have entered a command which belongs in another box, it will attempt to move you there. (This means that you can start typing an SQL query in the SELECT box, and, by including the SQL commands, prompt the ORDS to move you through the boxes as required. This may be a more efficient way of working for those who are already familiar with SQL.)

If you have a pre-written SQL query, you can paste this into the SELECT box, and the ORDS will separate it into its component parts.

Once you have entered your query, click Run query. Your query results will be displayed in a new browser tab. As before, your results can be sorted by clicking on the column headings.

5.2.1. SQL basics

Whilst this course book is not the place to give a full description of SQL, the basic commands that you might wish to use in the ORDS are fairly straightforward.

- **SELECT** statements indicate from which field(s) data should be returned
- **FROM** statements indicate the table(s) in which those fields are to be found
- **JOIN / ON** statements are used when you wish to query multiple tables
- **WHERE** statement provide the conditions that a record must meet in order to be listed in results; **AND** and **OR** can be used for advanced queries
- **ORDER BY** statements control the order in which results are returned

In order to return particular columns from a given table, you would write a query such as:

```
SELECT field1, field3, field4
FROM table1
```

If you only wanted to return records which met a particular criteria (such as the data in one field being greater than 100), then your query would look something like this:

```
SELECT field1, field3, field4
FROM table1
WHERE field4>100
```

If you wish to combine results from multiple tables, you need to indicate which tables you are drawing information from and how the tables are related. It’s common practice to use a form of shorthand to indicate which table each field belongs to, for instance:

```
SELECT a.field1, a.field3, a.field4, b.field1
FROM table1 a JOIN table2 b ON a.field2 = b.field2
WHERE a.field4>100
```

The letters after the table names in the FROM/JOIN part of the query are placed before the field names (followed by a full stop) to indicate which table the fields belong to. JOIN statements go in the FROM box on the SQL Query Page.

If any of your table names or field names include spaces, you will need to enclose these in double quotation marks, so that the ORDS recognizes the name as a single entity. For instance:

```
SELECT "field one", "field two" FROM "my table"
```
If your SQL query is not properly formed, you’ll just get an error message. You can’t delete data from the ORDS by simply entering a ‘wrong’ query.

5.3. Exporting query results

Whichever query tool you use, you have the option of exporting the results of your query as a .csv file. This allows them to be viewed outside the ORDS, or to be imported into other software for analysis.

To export the results of your query, click **Export query results as CSV**.

You can save the results, or, depending on the default settings of your computer, they may be opened automatically in a spreadsheet.

5.4. Saving query results as datasets

Project owners, project administrators, and contributors can also save their query results within the ORDS. In ORDS terminology, this is known as a ‘dataset’.

*You cannot save queries in trial project – you will need to apply for a full project first.*

In order to create a saved query, click **Save query results as dataset** after running a query. This will open the **Save Query** dialogue:

Similarly to the Create New Project and Add Database forms, this form allows you to give your new dataset a name and description, as well as specifying who can access it, and whether it is static or dynamic.

A key feature of the ORDS is that you can share or publish datasets even whilst keeping your underlying database private. If you **Save** your dataset with the Visibility set to public then people will be able to search for it and view it without needing to be logged in to the ORDS. Each public dataset is given its own unique URL so that it can be cited online or in publications. (However, you should note that there is no guarantee that ORDS datasets will remain available indefinitely.)
after your project has concluded: if you wish a dataset to be available long-term, you will need to make arrangements for it to be properly archived. Please contact the ORDS team by emailing ords@it.ox.ac.uk if you would like to discuss possible ways of doing this.)

Ticking the Create as static dataset box will result in your dataset being saved as a 'static' dataset, meaning that the dataset will be permanently fixed in the form it was when it was saved. If the underlying data later changes, these changes will not be reflected in the dataset. If the box is unchecked then the dataset will be saved as 'dynamic'. If the data in the underlying database is later changed, these changes will also appear in the saved dataset.

If you are planning on making the dataset public – to reference in a presentation or publication, for example – it is generally wise to save it as static. If you are saving it for other project members to refer to in future, then a dynamic dataset may be preferable.

Once you have created and saved your dataset, it will be added to the list of datasets belonging to your database. To access this list, click the View previously saved datasets button on the View, Edit, and Query Data page.

This will take you to the Dataset List for the current database:

The Dataset List shows all saved datasets in the current database and enables project members to re-run them, and, if they have the appropriate permissions, edit or delete them. The list also shows at a glance whether the dataset is viewable by the public, and whether it is static or dynamic.

**Exercise 6: Querying a database**

Now look at this exercise (page 71).
Exercise 7: Creating a new database from scratch

A final exercise drawing together a number of processes covered in this course book (page 75).
6 Connecting to the ORDS from third-party software and websites

The ORDS is a service that both hosts databases and provides online tools for viewing, editing, and querying the data it hosts. You may, however, wish to use the ORDS as a secure database hosting platform without using the online tools. You may, for instance, wish to edit your database using customized data entry forms that you have designed in Microsoft Access or other desktop software, or perhaps you have a project website and would like people to be able to access the information hosted by the ORDS via that website.

This is possible in the ORDS via the use of an ODBC (Open DataBase Connectivity) connection.

ODBC connections currently have to be set up manually by the ORDS team. However, work is underway to automate this process, and provide users with a straightforward way of doing this themselves, via the ORDS web interface. Additional documentation explaining how to do this will be provided once the new ODBC service is launched.

In the meantime, for further information about the ORDS and ODBC connections, please email ORDS team (ords@it.ox.ac.uk).
7 What next?

Courses offering training in database management software, database design and related topics are described below. In all cases, please refer to the IT Learning Programme web page (via www.it.ox.ac.uk/courses/) for further details.

7.1. Database design courses

Databases: Concepts of database design
Historical Research: Databases for research in History and Archaeology

7.2. Relational database management systems courses

Databases: Building a database using Access
Databases: Creating a user interface using Access
Databases: MySQL introduction
Databases: MySQL further techniques
Databases: Querying and analysing data using Access
Databases: Reporting data using Access

7.3. Course Clinic

We encourage everyone to work at their own pace. This may mean that you don’t manage to finish all of the exercises for this session. If this is the case, and you would like to complete the exercises while someone is on hand to help you, come along to one of the Course Clinic sessions that run during term time. More details are available from www.it.ox.ac.uk/courses/

7.4. Further help

It may be possible for you to use ITLP facilities in IT Services, Banbury Road to work through the exercises in this booklet after the course, or when you don’t have access to the applications on your own computer. Contact us on courses@it.ox.ac.uk for details.

The IT Services Help Centre is a good place to get advice about any aspect of using computer software or hardware. For Help Centre opening times, visit www.it.ox.ac.uk/help/gettinghelp/ and follow links to the General Helpdesk, or contact them by email on help@it.ox.ac.uk.

7.5. Downloadable course materials and more – the ITLP Portfolio

These course materials are available through the ITLP Portfolio, at http://portfolio.it.ox.ac.uk.

Each course pack includes the course handbook in PDF form and a zip folder of the exercise files that you need to complete the exercises. Archive versions of the course book may also be useful if you use an earlier version of the software.

The ITLP Portfolio helps you find articles, videos, resources and weblinks for further IT study. For some resources, you will be asked for your Oxford (SSO) username and password.
It may be possible for you to use the facilities at IT Services to work through the exercises in this booklet, or use any of the applications that are available. Contact us on courses@it.ox.ac.uk for details.

7.6. Useful websites

The main ORDS website is at http://ords.ox.ac.uk/.

The ORDS application is at https://app.ords.ox.ac.uk/.
## Appendix: Student Exercises

### Exercise 1  Registering to use the ORDS

- *Open a browser and navigate to the ORDS website*
- *Log in to the ORDS*
- *Register with the ORDS system, and verify your account*

#### Task 1

**Open the ORDS website**

#### Step 1

Start the computer if necessary.

#### Step 2

Click the `Start` button on the `Task Bar` at the bottom of the screen.

#### Step 3

If *Mozilla Firefox* appears on the `Start` menu, double-click on it.
Otherwise, start to type *Firefox* in the `Search programs and files` box, and double-click the entry when it appears.

(You can use a different browser if you prefer.)

#### Step 4

Type `http://staging.ords.ox.ac.uk` in the address bar, then press `ENTER`.

(The URL above is for the ORDS staging environment — a version of the system used for training and testing. If you are using the ORDS outside a training course, please use the main version: `http://app.ords.ox.ac.uk`.)

#### Task 2

**Log in to the ORDS**

#### Step 1

When the ORDS home page appears, click the `Log in using SSO` button.

#### Step 2

When the SSO log in screen appears, enter your username and password.

Click the `Login` button, then click the `✓ Continue` button on the confirmation page that follows.

(Your SSO username and password are those that you will use to access many University services, such as email, library resources, and WebLearn. Your SSO username will usually have the form `abcd1234`.

If you do not have an SSO username and password, please speak to the course instructor.)
<table>
<thead>
<tr>
<th>Task 3</th>
<th>Step 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Register with the ORDS system, and verify your account</td>
<td>If this is the first time you have used the ORDS, you will now be prompted to register. (If you have already registered, you will be taken straight to the ORDS welcome page.)</td>
</tr>
<tr>
<td></td>
<td>When the registration form appears, you will probably find that your details have been completed automatically. Check that your name and email address are appearing correctly, and edit them if necessary.</td>
</tr>
<tr>
<td></td>
<td>If the form has not been completed automatically, enter your name and your University email address.</td>
</tr>
<tr>
<td></td>
<td>When you are happy that your name and email address are correct, click the button.</td>
</tr>
<tr>
<td></td>
<td>Step 2</td>
</tr>
<tr>
<td></td>
<td>Clicking will trigger a confirmation email.</td>
</tr>
<tr>
<td></td>
<td>Open another browser tab, go to <a href="http://nexus.ox.ac.uk/">http://nexus.ox.ac.uk/</a>, and log in to your email account.</td>
</tr>
<tr>
<td></td>
<td>Find the ORDS confirmation email (this will have the subject line Message from ORDS) and open it. The email will contain a link; click on this to verify your ORDS account.</td>
</tr>
<tr>
<td></td>
<td>(If you do not see the confirmation email in your inbox, please check your junk email folder. If the message has not arrived within a few minutes of registering, please inform the course instructor.)</td>
</tr>
<tr>
<td></td>
<td>After clicking the link, you will be taken back to the ORDS welcome page, and should see a message confirming that your account has been verified.</td>
</tr>
<tr>
<td></td>
<td>Step 3</td>
</tr>
<tr>
<td></td>
<td>You will now have two tabs open showing the ORDS welcome page. Close one of these, but leave the other open for the next exercise.</td>
</tr>
</tbody>
</table>
## Exercise 2  Creating a new project

- *Create a new ORDS project*
- *Request full project status*

<table>
<thead>
<tr>
<th>Task 1</th>
<th>Create a project</th>
</tr>
</thead>
</table>
| **Step 1** | Go to the ORDS welcome page.  
(You should already have this on the screen from the previous exercise; if not, go to http://staging.ords.ox.ac.uk and log in again.) |
| **Step 2** | Click the [Create new project] button. |
| **Step 3** | When the Create New Project form appears, complete the Name and Description fields.  
Type My First ORDS Project (or another name of your choice) in the Name field.  
The Description field is intended to provide a summary of key information about the project. For now, type A project created during an ORDS training session in this field – you can return and edit this later if you choose.  
You can also complete the Start date and End date fields if you wish, but this is optional.  
Leave the Exclude from public ORDS project list and Enable ODBC access for project check boxes unticked. |
| **Step 4** | When you have completed the form, click [Create].  
You will be returned to the ORDS welcome page, where you should see a message saying Project created successfully.  
Underneath this will be a list of the ORDS projects you are currently involved in, including the project you have just created. |

<table>
<thead>
<tr>
<th>Task 2</th>
<th>Request full project status</th>
</tr>
</thead>
</table>
| **Step 1** | The project you have just created will be a trial project. However, some of the later exercises require you to have a full project.  
On the ORDS welcome page, click the name of the project you have just created.  
When the project page appears, scroll to the bottom of the page and click [Request full project]. This will trigger a full project request. You will be notified by email when the project has been upgraded. |
### Exercise 3  Importing a database file

- Create an empty ORDS database
- Import an existing database file

<table>
<thead>
<tr>
<th>Task 1</th>
<th>Step 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create an empty database</td>
<td>The project page for the project you created in Exercise 2 should still be on the screen. If not, go to the ORDS welcome page and click the project name. Click <strong>Add new ORDS database</strong>.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>When the Add Database to Project form appears, complete the Name and Description fields. In the Name field, type Victorian Gentlemen’s Clubs. In the Description field, type A database of members of London gentlemen’s clubs in the Victorian era.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click the small calendar icon at the right-hand end of the Created on field. A calendar will appear with today’s date highlighted in pink. Click the highlighted date to add the date to the field. Leave the other fields blank for the moment.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>When you are happy with the contents of the form, click <strong>Add</strong> to create the database. You will be taken back to the project page, and should see a message confirming that the database has successfully been created. The name of the new database will now appear in the list of ORDS databases for the project.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task 2</th>
<th>Step 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Import an existing database file</td>
<td>On the project page, click the name of the database you have just created.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>When the database page opens, click <strong>Upload existing database</strong>.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>When the Upload a database to ORDS page opens, click <strong>Browse...</strong>. (Note: if you are using a browser other than Firefox, the labelling of this button may vary.</td>
</tr>
</tbody>
</table>
Step 4
When the File Upload dialogue opens, navigate to Drive H:
Click the file called Clubs_database to select it, then click [Open].

Step 5
You will be returned to the Upload a database to ORDS page. You should now see the name of the file just selected alongside the Browse button.
Click [Upload].

Step 6
You will be returned to the project page, and should see two messages: one confirming that your database will be uploaded, and the other saying you will be notified by email when the process is complete.
Leave the project page on the screen for the next exercise.
If you log in to your email account, you should also see two messages from ORDS: one saying that the database has been sent to the servers for processing, and one confirming that the upload was successful.
(With large files, it may take some time for the second email to arrive. However, this database file is quite small, so should be processed quickly. If you have not received the second message after a few minutes, or if you receive a message saying there was a problem with the upload, please inform the course instructor.)
### Exercise 4  Viewing and editing the database structure

- View the database structure
- Add comments to database tables and fields
- Rename a field
- Add a new field to a table
- Save your changes and exit

<table>
<thead>
<tr>
<th><strong>Task 1</strong> View the database structure</th>
<th><strong>Step 1</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>The project page for the project created in <strong>Exercise 2</strong> should still be on the screen. If not, you can reach this from the ORDS welcome page by clicking the name of the project. Click the name of the database created in <strong>Exercise 3</strong>.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Task 2</strong> Add comments to database tables and fields</th>
<th><strong>Step 1</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Tables and fields in <strong>ORDS</strong> databases may have comments added to them, which can be used to record additional information. Let’s add a comment to the <strong>evidence</strong> table. Click the header of the <strong>evidence</strong> table to select it (the table border will change colour). Then click the <strong>Editable</strong> button in the toolbar on the right-hand side of the screen. When the <strong>Edit table</strong> dialogue box appears, type <strong>Details of the work in which evidence of club membership is found</strong> in the <strong>Comment</strong> field, then click <strong>OK</strong>.</td>
</tr>
</tbody>
</table>

| **Step 2** | View the comment by hovering the mouse pointer over the table header. |
### Step 3
Now let’s add a comment to the `end_date` field in the `addresses` table. Click the field to select it (two small red arrows will appear next to the field name).

Click the [Edit field] button in the toolbar on the right-hand side of the screen.

The `addresses` table will expand to show various additional options. Click [Edit comment].

In the dialogue box which appears, type **If club still at address, enter 2015.** Then click [OK]. The comment will appear at the bottom of the table.

Finally, click [Done] to shrink the `addresses` table back to its normal size.

### Step 4
View the comment by hovering the mouse pointer over the field.

Comments are also visible in the ORDS data editing interface, which will be used in a later exercise.

### Task 3
**Rename a field**

#### Step 1
Click the `first_name` field in the `people` table to select it (two small red arrows will appear next to the field name).

#### Step 2
Click the [Edit field] button in the toolbar on the right-hand side of the screen.

#### Step 3
The `people` table will expand to show various additional options. The field name will be highlighted.

Replace `first_name` with `forename`.

#### Step 4
Finally, click [Done] to shrink the `people` table back to its normal size.

### Task 4
**Add a new field to a table**

#### Step 1
Click the header of the `evidence` table to select it (the table border will change colour).

#### Step 2
Click the [Add field] button in the toolbar on the right-hand side of the screen.

#### Step 3
The `evidence` table will expand to show various additional options. A new field will have been added, with the placeholder name `new_field`. Replace the placeholder name with `publisher`. 
### Task 5  
**Save your changes and exit**  

<table>
<thead>
<tr>
<th>Step 1</th>
<th><strong>ORDS</strong> does not have an autosave function, so it is important to save your changes before leaving the schema design page. Click the <strong>Save</strong> button in the toolbar on the right-hand side of the screen. When the confirmation dialogue box appears, click <strong>OK</strong>.</th>
</tr>
</thead>
</table>
| Step 2 | When the **Changes saved** message appears, click **OK** to close the dialogue box.  
(If you experience any problems saving your database changes, please speak to the course instructor.) |
| Step 3 | To exit the schema design page, click the **Exit** button in the toolbar on the right-hand side of the screen. |
| Step 4 | Use the **Type** drop-down list to select the data type **Text**. |
| Step 5 | Click **Done** to shrink the **evidence** table back to its normal size. The new **publisher** field will turn pink, to indicate that it is a text field. |
**Exercise 5  Viewing and editing data**

- **View a table in an ORDS database**
- **Change the content displayed in a linked field**
- **Edit an existing record**
- **Apply filters to a table**
- **Add a new record**
- **Delete a record**
- **Export a copy of the database**

<table>
<thead>
<tr>
<th>Task 1</th>
<th>View a table</th>
<th>Step 1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>The database page for the database imported in <strong>Exercise 3</strong> should still be on the screen. If not, you can reach the database page from the ORDS welcome page by clicking the name of the project, and then the name of the database. Find the <strong>Main database version information</strong> pane (you may need to scroll down a little), and click <strong>View, edit, and query data</strong>.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>When the <strong>View, Edit, and Query Data</strong> page appears, you will see a list of the database tables in a grey pane on the left-hand side of the screen. Click <strong>addresses</strong> to view the contents of this table.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>The <strong>addresses</strong> table will appear. If necessary, use the scroll bars at the right-hand side and bottom of the viewing pane to see more of the contents.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task 2</th>
<th>Change the content displayed in a linked field</th>
<th>Step 1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>This table gives the addresses of the clubs listed in the database. It contains a field called <strong>club_id</strong>, which links to the <strong>clubs</strong> table. By default, this will show the content of the primary key field of the <strong>clubs</strong> table, which is called <strong>id</strong>. As this field simply contains a number, this isn’t very informative. However, we can change the field from the <strong>clubs</strong> table that is displayed. Underneath the field name is the <strong>Viewed field</strong> drop-down list. Click the small arrow at the right-hand end of this field to see the list of available fields. Click <strong>club_name</strong> to select it. The <strong>club_id</strong> field will now display the names of the clubs that had their premises at each address.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Return to the list of tables by clicking <strong>View, Edit, and Query Data</strong> in the breadcrumb trail towards the top of the screen.</td>
<td></td>
</tr>
<tr>
<td>Task 3</td>
<td>Step 1</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td></td>
</tr>
<tr>
<td>Edit an existing record</td>
<td>In the list of tables, click <strong>people</strong> to open that table.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Record number 5 of the <strong>people</strong> table contains two mistakes, and the date of death is also missing. Let’s correct this.</td>
</tr>
<tr>
<td>Click the <strong>title</strong> field for record number 5, and replace <strong>Mt</strong> with <strong>Mr</strong>.</td>
</tr>
<tr>
<td>Click the <strong>forename</strong> field, and replace <strong>Richrad</strong> with <strong>Richard</strong>. (If you did not change the field name in <strong>Exercise 4</strong> above, this will still be called <strong>first_name</strong>).</td>
</tr>
<tr>
<td>Click the <strong>death_year</strong> field (you may need to scroll across to see this), and replace <strong>[null value]</strong> with <strong>1906</strong>.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Note that the background of the fields which have been edited is blue, indicating unsaved changes.</td>
</tr>
<tr>
<td>Click <strong>Save changes</strong>. The cell backgrounds will revert to white.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task 4</th>
<th>Step 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apply filters to a table</td>
<td>With the <strong>people</strong> table still on the screen, scroll down to the filter interface at the bottom of the window.</td>
</tr>
<tr>
<td>From the first drop-down list, select <strong>forename</strong></td>
<td></td>
</tr>
<tr>
<td>From the second drop-down list, select <strong>is</strong></td>
<td></td>
</tr>
<tr>
<td>In the free-text field, type <strong>John</strong></td>
<td></td>
</tr>
<tr>
<td>Click <strong>Apply filter</strong>. Only those records whose subject’s first name is John will be displayed.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>To remove the filter and view all the records in the table, click <strong>Clear filter</strong>.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Now let’s apply a slightly more complex filter.</td>
</tr>
<tr>
<td>Click the blue plus symbol at the right-hand end of the filter interface to add a second row.</td>
</tr>
<tr>
<td>Ensure that the new drop-down list which appears to the left of the two rows is set to <strong>AND</strong></td>
</tr>
</tbody>
</table>
### Task 5
**Add a new record**

<table>
<thead>
<tr>
<th>Step 1</th>
<th>With the <strong>people</strong> table still on the screen, click <strong>Add new record</strong>. (The button is towards the bottom of the page: you may need to scroll down to see it.)</th>
</tr>
</thead>
</table>
| Step 2 | When the **Add New Record** form appears, complete the fields as follows:  
- **title**: Sir  
- **forename**: Algernon  
- **surname**: Ponsonby  
- **occupation**: Gentleman  
- **birth_year**: 1820  
- **death_year**: 1889  
- **notes**: Younger brother of Julian Ponsonby  
The **id** field will be completed automatically with the next available number. |
| Step 3 | Click **Save and return to table**.  
You will be returned to the table view, and should see the record you have just added at the bottom of the table. |

### Task 6
**Delete a record**

| Step 1 | Record number 7 is a duplicate of record number 6. Let’s tidy up the table by deleting it.  
Locate the record, and scroll to the right-hand end of the table. Click the **Delete** button at the end of record 7. |
### Task 7  
**Export a copy of the database**

<table>
<thead>
<tr>
<th><strong>Step 1</strong></th>
<th>Return to the database page by clicking <strong>Database Details</strong> in the breadcrumb trail towards the top of the screen.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 2</strong></td>
<td>Find the <strong>Main database version information</strong> pane (you may need to scroll down a little), and click <strong>Export database</strong>.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>When the export options page appears, use the drop-down list to select <strong>Single CSV file</strong>.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Click <strong>Export</strong>. When the file opening dialogue box appears, check that <strong>Microsoft Excel</strong> is selected as the program that will open the file (if not, use the drop-down list to select it), then click <strong>OK</strong>.</td>
</tr>
</tbody>
</table>
| **Step 5** | The file will open in **Excel**. Scroll down to see the contents of the various tables.  
To save a copy, click **File** and then **Save As**. When the **Save As** dialogue opens, navigate to **Drive H:**; then click **Save**. (You can now close this file if you wish.) |
| **Step 6** | Return to the browser tab displaying **ORDS**. Click **Return to database page**. |
## Exercise 6  Querying a database

- **Build a simple query using the query builder**
- **Write a more complex query using the SQL query page**
- **Save the query results as a dataset**

### Task 1
Build a simple query using the query builder

### Step 1
The database page for the database imported in **Exercise 3** should still be on the screen. If not, you can reach the database page from the **ORDS** welcome page by clicking the name of the project, and then the name of the database.

Find the **Main database version information** pane (you may need to scroll down a little), and click **View, edit, and query data**.

### Step 2
When the **View, Edit and Query Data** page appears, click **Query data using query builder**.

### Step 3
The query builder allows you to select one or more **ORDS** tables and see the contents of any or all of their fields. Let’s begin by creating a simple query for the **addresses** table.

Find the table in the **Tables** list (usually on the top right-hand side of the screen). Click the appropriate check box to select the **addresses** table.

### Step 4
The table will appear in the working area, displaying a list of the fields it contains and a check box next to each field name. Click all of the check boxes to select them.

An SQL query will appear in the **Query** box towards the bottom of the screen.

### Step 5
Click **Run query**. Your query results will be displayed in a new browser tab.

### Step 6
Notice that the **club_id** field in the query results only contains a number – the **id** number of the club record that each address is linked to.

To see which club is being referred to, we need to consult the **clubs** table. We can edit the query so that it does this. Start by going back to the browser tab displaying the query builder.
<table>
<thead>
<tr>
<th>Step 7</th>
<th>In the <strong>Tables</strong> list, click the box next to the <strong>clubs</strong> table. The table will appear in the working area, with a curved line showing the relationship between it and the <strong>addresses</strong> table. (You may need to move the tables to see the relationship clearly – you can do this by clicking a table header and dragging it.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 8</td>
<td>In the <strong>clubs</strong> table, click the check box next to the <strong>club_name</strong> field. We know that the <strong>club_id</strong> field in the <strong>addresses</strong> table just contains an uninformative primary key, so click the check box next to this field to deselect it. The SQL query in the <strong>Query</strong> box will update to reflect the new selections.</td>
</tr>
<tr>
<td>Step 9</td>
<td>Click <strong>Run query</strong>. The browser tab displaying query results will refresh – go to this tab to view the new results.</td>
</tr>
<tr>
<td>Step 10</td>
<td>Go back to the browser tab displaying the query builder, then return to the <strong>View, Edit, and Query Data</strong> page by clicking <strong>View, Edit, and Query Data</strong> in the breadcrumb trail towards the top of the screen.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task 2</th>
<th><strong>Write a more complex query using the SQL query page</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Click <strong>Query data using SQL</strong>.</td>
</tr>
</tbody>
</table>
| Step 2 | When the SQL query page appears, type the following:  
In the **SELECT** box:  
a.* , c.club_name  
In the **FROM** box:  
addresses a JOIN clubs c ON a.club_id = c.id  
Although it uses a slightly different syntax, this query is similar to the second one in the previous task – it will give the contents of the **addresses** table, but with the name of the clubs rather than just an **id** number. |
| Step 3 | Click **Run query**. Your query results will be displayed in a new browser tab. (If the browser tab containing the query results from the previous task is still open, this will refresh to display the new query results: go to this tab to see the results.) |
### Step 4

Go back to the tab displaying the SQL query page.

Now we’ll make the query into a conditional query. If we only want to see contemporary club addresses, we can add a **WHERE** statement to the above query so that it only returns results where the end date is later than 2010:

In the **WHERE** box, type:

```sql
a.end_date > 2010
```

An alternative method would be to put the cursor in the **SELECT** box, and then type in the whole query, including the SQL commands. **ORDS** will move you to the appropriate box as it detects that commands have been entered. In this case, the full query would look like this:

```sql
SELECT a.*, c.club_name FROM addresses a
JOIN clubs c ON a.club_id = c.id
WHERE a.end_date > 2010
```

### Step 5

Click the **Run query** button again. The query results tab will refresh to display the new results.

### Task 3

**Save the query results as a dataset**

### Step 1

We can now save the results of the query for future reference. In **ORDS** terminology, this is called creating a dataset.

(If you still cannot create datasets, please inform the course instructor.)

In the results tab, click **Save query results as dataset**.
### Step 2
When the **Save query** form appears, complete it as follows:

- **Dataset name**: Club addresses
- **Description**: List of addresses with associated club names, created on [today’s date].

(Replace *[today’s date]* in the above with the date on which you create the dataset.)

The **Query** field will be completed automatically.

Tick the **Create as static dataset** checkbox. This means the dataset will be a snapshot of the data at the point when it is saved – the contents will not be updated if the **addresses** or **clubs** tables are later edited.

From the **Visibility** drop-down list, select **Project members**. This means the dataset will only be visible to members of this **ORDS** project.

### Step 3
Click ![Save](#). You will be taken to the dataset list, and should see a message confirming that the dataset has been saved.

### Step 4
To view the dataset, click ![View dataset](#).

### Task 4
If you have time, you may like to return to the query builder and/or the SQL query page and experiment with constructing some other queries.

A brief guide to SQL statements is included in section 5.2.1 above (see page 52). For a more detailed guide, a useful SQL tutorial is available from W3Schools: [http://www.w3schools.com/sql/](http://www.w3schools.com/sql/)
## Exercise 7  Creating a new database from scratch

- Create an empty database
- Add tables and fields to the new database
- Add a relationship between tables
- View the new tables and add records

<table>
<thead>
<tr>
<th>Task 1</th>
<th>Create an empty database</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Return to the project page by clicking the project name in the breadcrumb trail towards the top of the screen.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Click <img src="image" alt="Add new ORDS database" /></td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>When the <strong>Add Database to Project</strong> form appears, complete the <strong>Name</strong> and <strong>Description</strong> fields.</td>
</tr>
<tr>
<td></td>
<td>In the <strong>Name</strong> field, type <strong>Bibliography</strong>.</td>
</tr>
<tr>
<td></td>
<td>In the <strong>Description</strong> field, type <strong>Details of useful source materials</strong>.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Click the small calendar icon at the right-hand end of the <strong>Created on</strong> field. A calendar will appear with today's date highlighted in pink.</td>
</tr>
<tr>
<td></td>
<td>Click the highlighted date to add it to the field.</td>
</tr>
<tr>
<td></td>
<td>Leave the other fields blank.</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>When you are happy with the contents of the form, click <img src="image" alt="Add" /> to create the database.</td>
</tr>
<tr>
<td></td>
<td>You will be taken back to the project page, and should see a message confirming that the database has successfully been created.</td>
</tr>
<tr>
<td></td>
<td>The name of the new database will now appear in the list of <strong>ORDS</strong> databases for the project.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task 2</th>
<th>Add tables and fields to the new database</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Click the name of the database you have just created to go to the database page.</td>
</tr>
<tr>
<td></td>
<td>Find the <strong>Main database version information</strong> pane (you may need to scroll down a little), and click <img src="image" alt="Create new database" /></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>You will see a blank schema design page. Click the <img src="image" alt="Add table" /> button on the toolbar. The mouse pointer will change from an arrow to a cross.</td>
</tr>
<tr>
<td></td>
<td>Click the place on the working area where you would like the table to appear. (You will be able to move it later if you wish.)</td>
</tr>
</tbody>
</table>
Step 3
The new table will appear, and the **Edit table** dialogue box will open. The table will have a placeholder name (beginning `new_table`) – replace this with `tblWorks`. (*tbl* is a prefix commonly used to denote tables in databases.)

Click **OK**.

You will see that your new table has been automatically populated with a field called `id`. This is the primary key field – the field that provides a unique identifier for each record.

Step 4
To add another field to the new table, click the table to select it (the table border will change colour), then click `Add field`.

Step 5
The table will expand to show various additional options. A new field will have been added, with the placeholder name `new_field`. Replace the placeholder name with `title`.

Step 6
Use the **Type** drop-down list to select the data type `Text`.

Step 7
Click **Done** to shrink the `tblWorks` table back to its normal size. The new `title` field will turn pink, to indicate that it is a text field.

Step 8
Repeat steps 4 to 7 to add three more fields to the table, with these names and data types:

- `author` (text)
- `publisher` (text)
- `date` (integer)

Note: the information entered in the `date` field will be a year, rather than a full date including date and month. Hence it is more appropriate to make this an `Integer` field, rather than using the `Date` data type.

Step 9
Repeat steps 2 and 3 to create a second table called `tblAuthors`. 

---

**ORDS: An Introduction**
**Step 10**
Repeat steps 4 to 7 to add three additional fields to the `tblAuthors` table:

- **name** (text)
- **birth_date** (integer)
- **death_date** (integer)

The completed tables should look like this:

(The data type labels next to each field name may not display by default. To see these, click `Options`, tick the **Show field data type in table design?** checkbox, then click `OK`.)

---

<table>
<thead>
<tr>
<th><strong>Task 3</strong></th>
<th><strong>Step 1</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a relationship between tables</td>
<td>We now need to join the two tables together, so that information from <code>tblAuthors</code> can be used to populate the <strong>author</strong> field in <code>tblWorks</code>.</td>
</tr>
<tr>
<td></td>
<td>We will do this by creating a relationship between the <strong>author</strong> field and the <strong>id</strong> field in <code>tblAuthors</code>. Fields that are linked in this way must have the same data type: this means that (perhaps counterintuitively), the <strong>author</strong> field needs to have the data type <code>Integer</code>.</td>
</tr>
<tr>
<td></td>
<td>Click the <strong>author</strong> field to select it, then click <code>Edit field</code>.</td>
</tr>
<tr>
<td></td>
<td>Use the <strong>Type</strong> drop-down list to select the data type <strong>Integer</strong>.</td>
</tr>
<tr>
<td></td>
<td>Click [Done]. The author field will now be yellow, indicating that it has a numeric data type.</td>
</tr>
</tbody>
</table>

**Step 2**
To create the relationship, click the **id** field of `tblAuthors` to select it. This will activate the `Connect foreign key` button on the toolbar; click this button. Now click the **author** field in `tblWorks`. A curved line should appear between the two fields, indicating that the relationship has been created.
### Task 4
**View the new tables and add records**

**Step 1**
From the database page, click **View, edit, and query data**.

**Step 2**
When the **View, edit, and query data** page appears, you will see the names of the two tables you have just created listed in a grey pane on the left-hand side of the screen.

Click **tblAuthors** to view this table.

**Step 3**
The table is currently empty, so no records will be displayed. Click **Add new record**.

**Step 4**
When the **Add New Record** form appears, complete the fields as follows:

- **name**: Anthony Fallon
- **birth_date**: 1923
- **death_date**: 2004

The **id** field will be completed automatically.

**Step 5**
Click **Save and return to table**.

You will be returned to the table view, and should see the record you have just added.

**Step 6**
Return to the list of tables by clicking **View, Edit, and Query Data** in the breadcrumb trail towards the top of the screen.

**Step 7**
Click **tblWorks** to view this table.

Once again, the table will be empty. Click **Add new record**.

---

**Step 3**
Click the **Save** button in the toolbar on the right-hand side of the screen. When the confirmation dialogue box appears, click **OK**.

Wait for the save operation to complete, then click **OK**. (If you experience any problems saving your database, please speak to the course instructor.)

To exit the schema design page, click the **Exit** button.
**Step 8**
When the **Add New Record** form appears, complete the fields as follows:

- **title**: The World of the London Club
- **publisher**: Carter and Sons
- **date**: 1967

**Step 9**
Because the **author** field is linked to the **tblAuthors** table, this requires a slightly different treatment. You will see two drop-down lists by this field, one on the left and one on the right.

Use the left-hand drop-down list to change the viewed field to **name**. This means that the right-hand drop-down list will display the contents of the **name** field, rather than record **id** numbers.

Now use the right-hand drop-down list to select **Anthony Fallon**. (The **tblAuthors** table currently only contains one record, so the list only has one name in it. If more records were added to **tblAuthors**, these would also appear in the drop-down list.)

**Step 10**
Click on **Save and return to table**.

You will be returned to the table view, and should see the record you have just added.

**Task 5**
If you have time, add some further records to the tables you’ve created.

You may also like to query the data using either the **Query Builder** or the **SQL Query Page**.