

INTERNATIONAL ADVANCED LEVEL

# INFORMATION TECHNOLOGY

Unit 4 - WIT14

SCHEME OF WORK

Pearson Edexcel International Advanced Subsidiary in Information Technology (XIT11)

Pearson Edexcel International Advanced Level in Information Technology (YIT11)

First teaching September 2018 First examination from June 2019

First certification from August 2019 (International Advanced Subsidiary) and August 2020 (International Advanced Level)



## INTRODUCTION

The following scheme of work provides an overview of the content of the 2018 International Advanced Level Information Technology and shows how the content could be taught as a guideline approach only.

It should be adapted by schools to fit their timetabling and staffing arrangements. It is based upon a two-year delivery model where all IAS content is being taught in the first year and the remaining IA2 content in the second year.

The scheme of work is broken up into units and topics, so that there is greater flexibility for moving topics around to meet planning needs.

It includes:

- Recommended teaching time for topics, though of course this is adaptable according to individual teaching needs
- Classroom activities, teaching points and suggested teaching resources
- Objectives for students at the end of the topic area and integrated Transferable Skills\* that are being developed.

The number of guided learning hours for Advanced Level is 360. Tutors should be aware that the estimated teaching hours are approximate and should be used as a guideline only.

## Unit 4

(Refer also to the [specification](#) and the [delivery and assessment guidance in the Getting Started Guide](#))

The scheme of work anticipates that students will have already completed these topics. Alternatively, tutors can add the topics to this scheme of work where appropriate.	
Unit 1 – Topic 5: Data and Databases	Unit 3 – Topic 12: Manipulating data
<ul style="list-style-type: none"> <li>5.2 Structured data</li> <li>5.3 Structured query language (SQL)</li> </ul>	<ul style="list-style-type: none"> <li>12.1 Data integrity</li> <li>12.2 Data normalisation</li> </ul>
Note	
<ul style="list-style-type: none"> <li>Topic 17 is not covered in isolation. It is automatically covered as part of the other topics. Other than the direct reference to 17.1 in Week 1 there are no references to this topic.</li> <li>Weeks 1 to 4 do not involve the creation of a database. They are designed to enable the students to learn the underpinning logic.</li> <li>Topic 19.4 has not been included as an isolated topic. It is covered as part of the other evaluation topics within the scheme of work.</li> <li>Whist teacher resources and student activities have been provided, the tutor could choose to replace these with their own resources/activities or edit them as they see fit.</li> <li>Solutions given are examples only. There are many ways of achieving the same outcome. Tutor/Students do not need to use these solutions.</li> <li>The teacher resources specified can be found <a href="#">here</a></li> <li>The student activities specified can be found <a href="#">here</a></li> </ul>	

Week	Topic area/aims/learning outcomes	Exemplar classroom activities/teaching points/suggested teaching resources	Integrated Transferable Skills
1	<p><b>17.1 Database applications</b> 17.1 Understand why database software is used to hold and manipulate data.</p> <p><b>18.2 Berrymill data structures</b> 18.2.1 Understand the need for and function of relational data structures.</p> <p><b>18.3 Data entry and validation</b> 18.3.1 Understand the need to ensure that stored data is suitable for processing and</p>	<p><b>Activity 1a: Why database software</b> Tutor to discuss reasons why database software is used to hold and manipulate data, for example:</p> <ul style="list-style-type: none"> <li>you can query (ask questions of) data in a database easily</li> <li>you can lookup up data</li> <li>you can relate data</li> <li>you can create meaningful reports</li> <li>they can handle very large data sets</li> <li>multiple users can use them.</li> </ul> <p>Students to work in pairs or small groups to discuss:</p> <ul style="list-style-type: none"> <li>databases they think they use</li> <li>databases they don't use but may hold data about them.</li> </ul> <p>Pairs/groups to report to whole class.</p>	<ul style="list-style-type: none"> <li>Communication</li> <li>Adaptive learning</li> <li>Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>

the methods used to achieve it.

**Resources**

- <https://www.bbc.co.uk/bitesize/guides/z8yg87h/revision/4>
- <https://www.liquidweb.com/blog/ten-ways-databases-run-your-life/>
- <https://www.quora.com/What-are-some-examples-of-databases-we-interact-with-every-day>

**Activity 1b: Why database software?**

Tutor to discuss impact of life without a database for a particular area, for example policing.

Students to work in pairs of small groups to discuss the impact on an area of life without databases. For example:

- education
- gaming
- social media
- health care
- library
- government

Pairs/groups to report to whole class.

**Resources**

- <https://www.interpol.int/en/How-we-work/Databases/Our-17-databases>
- [https://www.teach-ict.com/gcse\\_new/databases/uses\\_of\\_databases/miniweb/pg3.htm](https://www.teach-ict.com/gcse_new/databases/uses_of_databases/miniweb/pg3.htm)

**Activity 1c: Why database software?**

Tutor to discuss an example of a database that will hold data about the students, for example, school/college management information system (MIS).

Students to work in pairs of small groups to answer these questions:

1. Give examples of data that the database will hold
2. Give examples of queries/questions that could be needed
3. Give examples of reports that could be needed
4. Give examples of the users who would need to use the database

Pairs/groups to report to whole class.

**Activity 1d: Flat file databases**

Tutor to discuss/demonstrate a flat file database. Tutor could use the database provided in the tutor resources.

- Flat file database stores data in a single table
- Made up of columns and rows
  - column headings – attributes/fields
  - column – single item of data
  - row – tuple/record
- Data redundancy
- Anomalies
  - insert
  - update
  - delete
- Problems
  - storage
  - speed of access

Students to work individually or in pairs/small groups to complete Activity 1d.

**Resources**

- Tutor flat file database
- Student Activity 1d
- <https://study.com/academy/lesson/flat-file-database-definition-example.html>
- <https://www.defit.org/data-redundancy/>
- [https://databasemanagement.fandom.com/wiki/Category:Data\\_Anomalies](https://databasemanagement.fandom.com/wiki/Category:Data_Anomalies)
- <https://opentextbc.ca/dbdesign01/chapter/chapter-10-er-modelling/>

**Activity 1e: Introduction to Berrymill databases**

Tutor to discuss/demonstrate how a Berrymill database minimises data redundancy and eliminates anomalies. This is just an introduction so terminology can be kept to a minimum. Tutor could use the Berrymill database and the Berrymill school scenario provided in the tutor resources.

- Tables are related through keys
- Related data is stored in one table only
- Data in one table can be accessed from another through its key

Students to work individually or in pairs/small groups to complete Activity 1e.

**Resources**

- Tutor Berrymill database (BerrymillDatabase.accdb)
- Tutor Berrymill school scenario
- Student Activity 1e

<p>2</p>	<p><b>18.1 Structuring data</b>  18.1.1 Be able to construct and amend relational databases in terms of:</p> <ol style="list-style-type: none"> <li>tables</li> <li>records</li> <li>fields</li> <li>relationships.</li> </ol> <p>18.1.2 Be able to use appropriate data types when structuring data:</p> <ol style="list-style-type: none"> <li>text (limited length, unlimited length, memo)</li> <li>number (byte, integer, long integer, double, decimal)</li> <li>date/time</li> <li>currency</li> <li>Boolean (yes/no, on/off, true/false).</li> </ol> <p>18.1.3 Be able to format data types using <b>common</b> and customised formats.</p> <p><b>18.3 Data entry and validation</b>  18.3.1 Understand the need to ensure that stored data is suitable for processing and the methods used to achieve it.</p>	<p><b>Activity 2a: Database design. Structuring data</b>  Tutor to define and discuss:</p> <ul style="list-style-type: none"> <li>entity <ul style="list-style-type: none"> <li>name can be useful when naming tables</li> <li>becomes a record (row) in a table</li> </ul> </li> <li>attribute <ul style="list-style-type: none"> <li>name can be useful when naming fields</li> <li>becomes a field name (column heading)</li> </ul> </li> <li>relationships <ul style="list-style-type: none"> <li>1:M (one to many)</li> <li>M:N (many to many) cannot be implemented in a Berryhill database. A link/junction/intermediary table needs introducing to solve the M:N relationship.</li> <li>1:1 (one to one)</li> <li>keys <ul style="list-style-type: none"> <li>primary (uniquely identifies a record in the table, can only have 1 primary key per table)</li> <li>composite primary (combination of fields to uniquely identify a record in the table, can only have 1 composite primary key in a table)</li> <li>foreign (field in one table that is a primary key in another, can have multiple foreign keys in a table)</li> </ul> </li> <li>referential integrity <ul style="list-style-type: none"> <li>accuracy and consistency of data within a relationship</li> <li>e.g. changing a primary key in one table must cascade through to all relevant foreign keys</li> </ul> </li> </ul> </li> </ul> <p>Students to work individually to complete Activity 2a.  Students to check their work with another classmate and discuss any differences.  Tutor to discuss answers.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>Student Activity 2a</li> <li><a href="https://database.guide/the-3-types-of-relationships-in-database-design/">https://database.guide/the-3-types-of-relationships-in-database-design/</a></li> <li><a href="https://www.guru99.com/dbms-keys.html">https://www.guru99.com/dbms-keys.html</a></li> <li><a href="https://www.quora.com/What-is-the-difference-between-primary-key-composite-key-and-foreign-key">https://www.quora.com/What-is-the-difference-between-primary-key-composite-key-and-foreign-key</a></li> <li><a href="https://www.techopedia.com/definition/1233/referential-integrity-ri">https://www.techopedia.com/definition/1233/referential-integrity-ri</a></li> </ul>	<ul style="list-style-type: none"> <li>Communication</li> <li>Adaptive learning</li> <li>Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>
----------	--	---	--

**Activity 2b: Database design. Data types**

Tutor to define and discuss data types including:

- text
  - limited length (short text = 255 characters, can be limited further by specifying a field size)
  - unlimited length/memo
- number
  - byte
  - integer
  - long integer
  - double
  - decimal
  - AutoNumber
- date/time
- currency
- Boolean (yes/no, on/off, true/false).

Students to work individually to complete Activity 2b.

Students to check their work with another classmate and discuss any differences.

Tutor to discuss answers.

**Resources**

- Student Activity 2b
- <https://support.office.com/en-us/article/introduction-to-data-types-and-field-properties-30ad644f-946c-442e-8bd2-be067361987c>

**Activity 2c: Database design. Formats**

Tutor to define and discuss common formats (customised formats will be covered later in the scheme of work) including:

- numeric formats
  - general number
  - fixed (with and without decimal places)
  - standard
  - percent
  - currency (with and without decimal places)
- date and time formats
  - general date
  - long date
  - medium date
  - short date
  - long time

		<ul style="list-style-type: none"> <li>- medium time</li> <li>- short time</li> <li>• Boolean <ul style="list-style-type: none"> <li>- Yes/No</li> <li>- True/False</li> <li>- On/Off</li> </ul> </li> </ul> <p>Students to work individually to complete Activity 2c. Students to check their work with another classmate and discuss any differences. Tutor to discuss answers.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Student Activity 2c</li> <li>• <a href="https://support.office.com/en-us/article/introduction-to-data-types-and-field-properties-30ad644f-946c-442e-8bd2-be067361987c">https://support.office.com/en-us/article/introduction-to-data-types-and-field-properties-30ad644f-946c-442e-8bd2-be067361987c</a></li> </ul>	
3	<p><b>18.1 Structuring data</b> 18.1.3 Be able to format data types using common and <b>customised</b> formats.</p> <p><b>18.3 Data entry and validation</b> 18.3.1 Understand the need to ensure that stored data is suitable for processing and the methods used to achieve it.</p> <p>18.3.2 Be able to use validation techniques that can be used to ensure data accuracy:</p> <ol style="list-style-type: none"> <li>a. presence check</li> <li>b. range check</li> <li>c. lookup check</li> <li>d. list check</li> <li>e. format (picture) check</li> <li>f. length of data check</li> </ol> <p>18.3.3 Be able to construct appropriate error messages</p>	<p><b>Activity 3a: Length of data checks</b> Tutor to discuss and demonstrate field sizes. The Berrymill database and Berrymill school scenario could be used for this. To include:</p> <ul style="list-style-type: none"> <li>• only suitable for text fields</li> <li>• limits number of characters that can be input</li> <li>• only suitable for text fields</li> <li>• some text fields will have an exact length where there will never be more or less characters e.g. passport number UK should be a short text data type with 9 characters</li> <li>• some will have a best guess length e.g. first name</li> <li>• are not entirely user-friendly e.g.do not produce an error message if the user tries to input more characters</li> <li>• could be used in combination with other validation techniques e.g. length check as validation rule with a suitable error message</li> </ul> <p>Students to work individually to complete Activity 3a - Weekend Run Club. They will add to this for activities 3b to 3f.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Student Activity 3a - Weekend Run Club</li> <li>• Tutor Berrymill database</li> <li>• Tutor Berrymill school scenario</li> <li>• <a href="http://en.tekstenuitleg.net/articles/software/access-validation-rule-tutorial/list-of-access-validation-rules">http://en.tekstenuitleg.net/articles/software/access-validation-rule-tutorial/list-of-access-validation-rules</a></li> </ul>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Adaptive learning</li> <li>• Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>

that give users appropriate and helpful feedback as to the nature of the problem.

18.3.4 be able to use techniques to aid and improve the quality of data entry.

- a. user help
- b. input masks

**Activity 3b: Database design. Format (picture) checks and input masks**

Tutor to discuss and demonstrate format checks and input masks. The Berryhill database and Berryhill school scenario could be used for this. To include:

- input masks used to force the user to input data in a particular format
- ignored if data is imported
- pre-defined input masks
- custom input masks
- are not entirely user friendly e.g. do not produce error message
- could be used in combination with or replaced by other validation techniques e.g. validation rule and validation text with a suitable error message

Students to work individually to identify and specify suitable input masks in their Activity 3a – Weekend Rub Club.

**Resources**

- <https://support.office.com/en-us/article/control-data-entry-formats-with-input-masks-e125997a-7791-49e5-8672-4a47832de8da>

**Activity 3c: Database design/ Presence checks**

Tutor to discuss and demonstrate presence checks. The Berryhill database and Berryhill school scenario could be used for this. To include:

- ensure data has to be present
- why setting 'Required' to yes is not appropriate (error message is not user friendly)
- using a validation rule with validation text to ensure error message can be customised (Is Not Null)
- should only be used on fields where there has to be data when the record is created
- must include suitable error message
- could be used in combination with other validation techniques

Students to work individually to identify and specify suitable presence checks in their Activity 3a – Weekend Rub Club.

**Resources**

- Tutor Berryhill database
- Tutor Berryhill school scenario
- <https://support.office.com/en-us/article/control-data-entry-formats-with-input-masks-e125997a-7791-49e5-8672-4a47832de8da>

**Activity 3d: Database design. Range checks**

Tutor to discuss and demonstrate range checks. The Berryhill database and Berryhill school scenario could be used for this. To include:

- number ranges
- date ranges
- combinations of
  - greater than or equal to ( $\geq$ )
  - less than or equal to ( $\leq$ )
  - greater than ( $>$ )
  - less than ( $<$ )
  - BETWEEN lower limit AND upper limit
- must include suitable error message

Students to work individually to identify and specify suitable range checks in their Activity 3a – Weekend Rub Club.

**Resources**

- Tutor Berryhill database
- Tutor Berryhill school scenario
- <https://support.office.com/en-us/article/restrict-data-input-by-using-validation-rules-b91c6b15-bcd3-42c1-90bf-e3a0272e988d>

**Activity 3e: Database design. List check/value lookup check**

Tutor to discuss and demonstrate value lookups. The Berryhill database and Berryhill school scenario could be used for this. To include:

- only used where there is a limited range of options
- not used on foreign keys
- combo boxes
- list boxes
- limiting to list

Students to work individually to identify and specify suitable value lookups in their Activity 3a – Weekend Rub Club.

**Resources**

- Tutor Berryhill database
- Tutor Berryhill school scenario
- <https://support.office.com/en-us/article/control-data-entry-formats-with-input-masks-e125997a-7791-49e5-8672-4a47832de8da>

		<p><b>Activity 3f: Database design. Table lookup check (foreign keys)</b>  Tutor to discuss and demonstrate table lookup checks for foreign keys. The Berrymill database and Berrymill school scenario could be used for this. To include:</p> <ul style="list-style-type: none"> <li>• used on foreign keys to provide combo box of options</li> <li>• limit to list specified to yes</li> </ul> <p>Students to work individually to identify and specify suitable table lookups in their Activity 3a – Weekend Rub Club.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Berrymill database</li> <li>• Tutor Berrymill school scenario</li> </ul>	
4	<p><b>18.1 Structuring data</b></p> <p><b>18.3 Data entry and validation techniques</b></p> <p><b>19.1 Using database software</b></p> <p>19.1.6 Be able to import data from external sources: .txt</p>	<p><b>Activity 4: Database design. Consolidation</b>  Tutor to discuss methods of studying a scenario and dataset. Tutor could use Berrymill school scenario and Berrymill data set for this. For example:</p> <ul style="list-style-type: none"> <li>• studying scenario to determine <ul style="list-style-type: none"> <li>– possible tables</li> <li>– possible fields and the tables they belong in</li> <li>– possible validation</li> </ul> </li> <li>• import data set(s) to study <ul style="list-style-type: none"> <li>– possible tables</li> <li>– possible fields and the tables they belong in</li> <li>– possible validation</li> </ul> </li> <li>• complete table designs for each table</li> </ul> <p>Students to work individually to complete Activity 4 - Wallsherpool</p> <ul style="list-style-type: none"> <li>• study a scenario and data set</li> <li>• determine <ul style="list-style-type: none"> <li>– entities/tables</li> <li>– fields/attributes</li> <li>– primary keys</li> <li>– foreign keys</li> <li>– relationships</li> <li>– data types</li> <li>– validation including (where suitable): <ul style="list-style-type: none"> <li>○ presence checks</li> <li>○ range checks</li> <li>○ length of data check</li> <li>○ format (picture) check including input masks</li> </ul> </li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Adaptive learning</li> <li>• Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>

		<ul style="list-style-type: none"> <li>○ table lookup check</li> <li>○ value list check</li> <li>○ suitable user help in the form of error message</li> </ul> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Berrymill database</li> <li>• Tutor Berrymill school scenario</li> <li>• Student Activity 4 - Wallsherpool</li> </ul>	
5	<p><b>18.1 Structuring data</b></p> <p><b>18.2 Relational data structures</b></p> <p>18.2.2 be able to implement the features and attributes of data relationships in terms of:</p> <ol style="list-style-type: none"> <li>a. relationships (many-to-one, many-to-many, one-to one)</li> <li>b. primary, foreign and composite keys</li> <li>c. referential integrity</li> </ol> <p><b>18.3 Data entry and validation techniques</b></p> <p><b>19.1 Using database software</b></p> <p>19.1.1 Be able to update, insert, modify and delete data</p> <p>19.1.2 Be able to create Berrymill data structures to handle given data sets</p> <p>19.1.6 Be able to import data from external sources: .txt</p>	<p><b>Activity 5: Database implementation. Building tables and relationships</b></p> <p>Tutor to demonstrate/discuss constructing a Berrymill database. There are a number of different methods. This is one example:</p> <ul style="list-style-type: none"> <li>• data set(s) imported into Access</li> <li>• imported data set copied and pasted (structure only) for each table that is required</li> <li>• fields not required deleted from each table</li> <li>• data types assigned</li> <li>• primary keys assigned</li> <li>• validation assigned</li> <li>• referential integrity enforced</li> <li>• use append queries to append data from data set(s) into relevant tables (always the 1 side of the relationship before the M side of the relationship)</li> </ul> <p>Students to work individually to build a database based on their completed Activity 4 - Wallsherpool.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Student Activity 4 - Wallsherpool</li> <li>• Student Creating a database</li> </ul>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Adaptive learning</li> <li>• Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>

6	<p><b>18.1 Structuring data</b></p> <p><b>18.2 Relational data structures</b></p> <p><b>18.3 Data entry and validation techniques</b></p> <p><b>19.1 Using database software</b></p>	<p><b>Activity 6: Database implementation: Building tables and relationships. Consolidation Activity</b> Tutor to facilitate database structure consolidation activity.</p> <p>Students to work individually to build a database from Activity 6 – Cinema Club scenario and data sets. <b>Note:</b> the table design template has been taken out of the activity. Students are not tested on their ability to complete a table design template.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Student Activity 6 - Cinema Club</li> </ul>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Adaptive learning</li> <li>• Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>
		<p><b>Activity 7: Database implementation: Building tables and relationships. Consolidation Activity</b> Tutor to facilitate database structure consolidation activity.</p> <p>Students to work individually to build a database from Activity 7 – Raptshia scenario and data set.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Student Activity 7 - Raptshia</li> </ul>	
7	<p><b>19.1 Using database software</b></p> <p>19.1.3 Be able to retrieve data for specific purposes: queries</p> <p>19.2.2 Be able to create system outputs for a database solution that aid users effectively: data tables</p> <p>19.1.4 Be able to construct calculated fields in queries and reports to generate meaningful information and solve problems</p> <p>19.1.5 Be able to output data in appropriate formats</p>	<p><b>Activity 8: Queries</b> Tutor to discuss/demonstrate query criteria. Tutor could use Berryhill school scenario and Berryhill data set for this. For example:</p> <ul style="list-style-type: none"> <li>• exact match e.g. student with passport number MQ2416179</li> <li>• Boolean e.g. all the students who do not yet have parental consent</li> <li>• equal to (=) e.g. all the students who chose to pay in 1 instalment</li> <li>• sort single ascending/descending</li> <li>• wildcards e.g. <ul style="list-style-type: none"> <li>– all the students with postcodes beginning with FE</li> <li>– all the students with addresses that include Road</li> </ul> </li> <li>• greater than/greater than or equal to (&gt;, &gt;=) e.g. <ul style="list-style-type: none"> <li>– all students who have chosen to pay in more than 1 instalment</li> <li>– all the students who were born on or after 22/04/2003</li> </ul> </li> <li>• less than/less than or equal to (&lt;, &lt;=, BETWEEN AND) e.g. <ul style="list-style-type: none"> <li>– all the students who have chosen to pay in less than 3 instalments</li> <li>– all the students who were born before 22/04/2003</li> </ul> </li> <li>• OR e.g. <ul style="list-style-type: none"> <li>– all the students who have chosen to pay in 2 or 3 instalments</li> <li>– all the students whose emergency contact is either a father or grandfather</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Adaptive learning</li> <li>• Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>

<p>to suit intended purpose and user.</p>	<ul style="list-style-type: none"> <li>• AND e.g. <ul style="list-style-type: none"> <li>– all the students who have parental consent and whose emergency contact is their guardian</li> <li>– all the students who are visiting the Eiffel Tower on the 15/03/2020 (demonstrate using more than one table)</li> </ul> </li> <li>• NOT e.g. <ul style="list-style-type: none"> <li>– All the students who have chosen not to visit the Eiffel Tower on the 15/03/2020</li> </ul> </li> </ul> <p>Students to work in pairs using their Wallsherpool database (completed in activities 4 and 5) to complete Activity 8 – Wallsherpool Queries. The database has also been provided in the tutor resources folder.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Berryhill database</li> <li>• Student Activity 8 – Wallsherpool Queries</li> <li>• Tutor Activity 8 – Wallsherpool Queries database</li> </ul>	
	<p><b>Activity 9: Aggregate Function Queries</b></p> <p>Tutor to demonstrate/discuss aggregate function queries. Tutor could use Berryhill school scenario and Berryhill data set for this. For example:</p> <ul style="list-style-type: none"> <li>• naming calculated fields (name of new field:followed by existing name of field)</li> <li>• min e.g. lowest payment amount made</li> <li>• max e.g. highest payment amount made</li> <li>• count e.g. number of payments made</li> <li>• sum e.g. sum of payments made</li> <li>• avg e.g. average of payments made. Force display to 2 decimal places</li> <li>• where e.g. count of visits to the Eiffel Tower (included in Berryhill database)</li> </ul> <p>Note the queries have been provided in the Berryhill database.</p> <p>Students to work individually using their Wallsherpool database to complete Activity 9 – Wallsherpool Queries. The database has also been provided in the tutor resources folder.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Berryhill database</li> <li>• Student Activity 9 – Wallsherpool Queries</li> <li>• Tutor Activity 9 – Wallsherpool Queries database</li> </ul>	

		<p><b>Activity 10: Calculated Queries</b>  Tutor to demonstrate/discuss calculated queries. Tutor could use Berry mill school scenario and Berry mill data set for this. For example:</p> <ul style="list-style-type: none"> <li>• addition e.g. <ul style="list-style-type: none"> <li>– total amount that has already been paid for the trip plus £1200 for the three teachers who are going on the trip.</li> </ul> </li> <li>• subtraction e.g. <ul style="list-style-type: none"> <li>– payment amount outstanding for students who have not paid the full amount</li> </ul> </li> <li>• multiplication e.g. <ul style="list-style-type: none"> <li>– total amount of insurance paid plus 3 x £15 for the three teachers who are going on the trip</li> </ul> </li> <li>• division e.g. <ul style="list-style-type: none"> <li>– thinking of changing instalments so that equal amount has to be paid in each. Query to find out instalment amount.</li> </ul> </li> <li>• dates e.g. <ul style="list-style-type: none"> <li>– age of students</li> <li>– days since visit to Eiffel Tower on the 15/03/2012</li> <li>– weeks since visit to Eiffel Tower on 15/03/2012</li> <li>– months since visit to Eiffel Tower on the 15/03/2012</li> <li>– years since visit to Eiffel Tower on the 15/03/2012</li> </ul> </li> </ul> <p>Note the queries have been included in the Berry mill database.</p> <p>Students to work individually using their Wallsherpool database to complete Activity 10 – Wallsherpool Queries. The database has also been provided in the activities folders as a new table has been added.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• <a href="https://www.techonthenet.com/access/functions/date/datediff.php">https://www.techonthenet.com/access/functions/date/datediff.php</a></li> <li>• Tutor Berry mill database</li> <li>• Student Activity 10 – Wallsherpool Queries and Database</li> <li>• Tutor Activity 10 – Wallsherpool Queries database</li> </ul>	
8	<p><b>19.1 Using database software</b></p> <p>19.1.3 Be able to retrieve data for specific purposes: queries</p>	<p><b>Activity 11: Conditional Queries (iif)</b>  Tutor to demonstrate/discuss iif statements in queries. Tutor could use Berry mill school scenario and Berry mill data set for this. For example:</p> <ul style="list-style-type: none"> <li>• iif statements e.g. <ul style="list-style-type: none"> <li>– if a student is going to pay in 1 instalment then they will get a 10% discount on the £400 cost of the trip.</li> </ul> </li> </ul>	

<p>19.1.4 Be able to construct calculated fields in queries and reports to generate meaningful information and solve problems</p> <p>19.1.5 Be able to output data in appropriate formats to suit intended purpose and user.</p> <p><b>19.2 User interface</b></p> <p>19.2.2 Be able to create appropriate system outputs for a database solution that aids users effectively: reports</p>	<ul style="list-style-type: none"> <li>- If a student has not yet paid £400 then display standard payment message "You must pay the balance within 7 days" otherwise display "You have fully paid – thank you")</li> <li>- two column query only; number of students aged 16, number of students aged 17.</li> </ul> <p>Note the queries have been included in the Berryhill database.</p> <p>Students to work individually using their Wallsherpool database to complete Activity 11 – Wallsherpool Queries. The database has also been provided in the tutor resources folder.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Berryhill database</li> <li>• Student Activity 11 – Wallsherpool Queries</li> <li>• Tutor Activity 11 – Wallsherpool Queries database</li> </ul> <hr/> <p><b>Activity 12: Parameter Queries</b></p> <p>Tutor to demonstrate/discuss parameter queries. Tutor could use Berryhill school scenario and Berryhill data set for this. For example:</p> <ul style="list-style-type: none"> <li>• display the student surname and total amount paid for a student surname input as a parameter</li> <li>• display the number of students who are paying the cost of their trip within a range (input as parameters) of instalments.</li> </ul> <p>Note the queries have been included in the Berryhill database.</p> <p>Students to work individually using their Wallsherpool database to complete Activity 12 – Wallsherpool Queries. The database has also been provided in the tutor resources folder.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Berryhill database</li> <li>• Student Activity 12 – Wallsherpool Queries</li> <li>• Tutor Activity 12 – Wallsherpool Queries database</li> </ul>	
--	--	--

		<p><b>Activity 13: Action Queries</b> Tutor to demonstrate/discuss action queries. Tutor could use Berry mill school scenario and Berry mill data set for this. For example:</p> <ul style="list-style-type: none"> <li>• append queries <ul style="list-style-type: none"> <li>– if not already covered as part of importing data and creating table structures tutors may introduce here or leave until forms are covered as they will be used again there</li> </ul> </li> <li>• update queries e.g. <ul style="list-style-type: none"> <li>– the student county needs updating from County Durham to Tyne and Wear</li> <li>– all of the students have now paid their £15.00 travel insurance. The insurance amount for those currently at £0 needs updating to £15.00</li> </ul> </li> <li>• delete queries e.g. <ul style="list-style-type: none"> <li>– Philip Teneur and David Popal are no longer going on the trip. They need to be deleted</li> </ul> </li> </ul> <p>Note the queries have been included in the Berry mill database.</p> <p>Students to work individually using their Wallsherpool database to complete Activity 13 – Wallsherpool Queries. The database has also been provided in the tutor resources folder.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Berry mill database</li> <li>• Student Activity 13 – Wallsherpool Queries</li> <li>• Tutor Activity 13 – Wallsherpool Queries database</li> </ul>	
9	<p><b>19.1 Using database software</b></p> <p>19.1.3 Be able to retrieve data for specific purposes: reports</p> <p>19.1.4 Be able to construct calculated fields in queries and reports to generate meaningful information and solve problems</p> <p>19.1.5 Be able to output data in appropriate formats</p>	<p><b>Activity 14 – Ungrouped Reports</b> Tutor to demonstrate/discuss reports. Tutor could use Berry mill school scenario task 4 and the Berry mill database. For example:</p> <ul style="list-style-type: none"> <li>• single table report using the report wizard based on <b>qryAmountOutstanding</b></li> <li>• bound controls – control whose source of data is a field in a table or query</li> <li>• calculated control – control whose source of data is an expression. For example, a SUM</li> <li>• unbound control – control that does not have a field or expression as its source. For example, a line or shape.</li> <li>• report sections <ul style="list-style-type: none"> <li>– report header <ul style="list-style-type: none"> <li>○ appears once in the report at the top of the first page</li> </ul> </li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Adaptive learning</li> <li>• Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>

to suit intended purpose and user.

## 19.2 User interface

19.2.2 Be able to create appropriate system outputs for a database solution that aids users effectively: reports

- use for calculations that apply to a field or fields for the entire report that you want to appear here
- report footer
  - appears once at the end of the report on the last page
  - use report footer for calculations that apply to a field or fields for the entire report that you want to appear here
- page header
  - appears after the report header on the first page and then at the top of each subsequent page.
  - do not use for calculations
  - field labels belong here (unless using grouping which will be discussed later)
- page footer
  - appears at the bottom of every page and above the report footer on the last page.
  - do not use for calculations
- detail section
  - printed once for every row in the record source
  - can add calculations for each row if required
- formatting the report
  - useful title
  - useful labels
  - use of white space
  - setting report header colour to white
  - setting alternate row background colour to white
  - adding calculation controls in report footer
    - suitable generated field names
    - SUM formula
    - formatting to currency with 2 decimal places
  - conditional formatting
    - background colour of AmountOutstanding should be red if the student has not fully paid
  - useful tools
    - align
    - size/space

Note the report has been included in the Berryhill database (rptAmountOutstanding).

Students to work individually to complete Activity 14 – Wallsherpool Reports. The database has been provided for the students. The database has also been provided in the tutor resources folder.

### Resources

- Tutor Berrymill database
- Tutor Activity 14 – Wallsherpool Reports database
- Student Activity 14 – Wallsherpool Reports and Database
- <https://support.office.com/en-us/article/create-a-simple-report-408e92a8-11a4-418d-a378-7f1d99c25304> (creating simple reports)
- [https://www.youtube.com/watch?v=j313-eqzu\\_Q](https://www.youtube.com/watch?v=j313-eqzu_Q) (conditional formatting)
- <https://support.office.com/en-us/article/examples-of-expressions-d3901e11-c04e-4649-b40b-8b6ec5aed41f> (calculations)

### Activity 15 – Grouped Reports

Tutor to demonstrate/discuss reports. Tutor could use Berrymill school scenario task 5 and Berrymill database for this. For example:

- query to generate the data for the report including:
  - concatenation of surname and forename
  - wildcard to search for Eiffel Tower trips
  - SUM to generate total amount paid
- report based on the query
  - grouped by description
  - description header includes:
    - description and visit date fields
    - description, visit date, student name, total amount paid labels
  - detail section includes student name and total amount paid fields
  - description footer includes calculations for:
    - overall amount paid
    - average amount paid
    - total students
  - report footer includes calculations for:
    - overall amount paid
    - average amount paid
    - total students
- report is formatted:
  - useful title
  - useful labels
  - use of white space
  - setting report header colour to white
  - setting alternate row background colour to white
  - suitable generated field names
  - formatting to currency with 2 decimal places

Note the report has been included in the Berrymill database (rptEiffelTower).

		<p>Students to work individually to complete Activity 15 – Wallsherpool Reports. The database has been provided for the students. The database has also been provided in the tutor resources folder.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Berryhill database</li> <li>• Tutor Activity 15 – Wallsherpool Reports database</li> <li>• Student Activity 15 – Wallsherpool Reports and Database</li> <li>• <a href="https://www.quackit.com/microsoft_access/microsoft_access_2016/how_to/how_to_group_a_report_in_access_2016.cfm">https://www.quackit.com/microsoft_access/microsoft_access_2016/how_to/how_to_group_a_report_in_access_2016.cfm</a> (basic grouping)</li> <li>• <a href="https://support.office.com/en-us/article/examples-of-expressions-d3901e11-c04e-4649-b40b-8b6ec5aed41f">https://support.office.com/en-us/article/examples-of-expressions-d3901e11-c04e-4649-b40b-8b6ec5aed41f</a> (calculations)</li> </ul>	
10-11	<p><b>19.1 Using database software</b></p> <p>19.1.3 Be able to retrieve data for specific purposes: reports</p> <p>19.1.4 Be able to construct calculated fields in queries and reports to generate meaningful information and solve problems</p> <p>19.1.5 Be able to output data in appropriate formats to suit intended purpose and user.</p> <p><b>19.2 User interface</b></p> <p>19.2.2 Be able to create appropriate system outputs for a database solution that aids users effectively: charts</p>	<p><b>Activity 16: Single Series Column Charts</b></p> <p>Tutor to demonstrate/discuss column charts. Tutor could use Berryhill school scenario task 6 and Berryhill database for this. For example:</p> <ul style="list-style-type: none"> <li>• query to generate data for the chart including COUNT of students for each instalment type</li> <li>• create report using ‘report design’ option</li> <li>• using the ‘insert chart’ option</li> <li>• selecting column chart type</li> <li>• specifying a query as a data source</li> <li>• chart property sheet <ul style="list-style-type: none"> <li>– legend properties – no legend on single series charts</li> <li>– chart title – suitable title. Larger font than rest of chart</li> <li>– category axis title. suitable title, suitable font size</li> <li>– primary value axis title, suitable title, suitable font size</li> <li>– primary value axis format (used to display numbers of different formats. Does not need changing in this example but worth demonstrating)</li> </ul> </li> <li>• changing page layout of report to landscape</li> <li>• ensuring report uses full size of page</li> </ul> <p>Note the chart has been included in the Berryhill database (rptChart1).</p> <p>Students to work individually to complete Activity 16 – Wallsherpool Charts. The database has been provided for the students. The database has also been provided in the tutor resources folder.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Berryhill database</li> <li>• Tutor Activity 16 – Wallsherpool Charts database</li> <li>• Tutor Creating Single Series Column Charts</li> <li>• Student Activity 16 – Wallsherpool Charts and Database</li> </ul>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Adaptive learning</li> <li>• Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>

### **Activity 17: Multi Series Column Charts**

Tutor to demonstrate/discuss column charts. Tutor could use Berryhill school scenario task 7 and Berryhill database for this. For example:

- query to generate data for the chart including
  - concatenated forename and surname
  - SUM payment amount < 400
  - MAX payment amount
  - MIN payment amount
- create report using 'report design' option
- using the 'insert chart' option
- selecting the column chart type
- specifying a query as a data source
- chart setting options
  - data
    - values (Y axis) needs all three series selecting
  - format
    - display name – needs to be suitable for each series
- chart property sheet
  - legend required, suitable font size
  - chart title – suitable title. Larger font than rest of chart
  - category axis title. suitable title, suitable font size
  - primary value axis title, suitable title, suitable font size
  - primary value axis format (change to fixed to show decimal places)
- changing page layout of report to landscape
- ensuring report uses full size of page

Note the chart has been included in the Berryhill database (rptChart2).

Students to work individually to complete Activity 17 – Wallsherpool Charts. The database has been provided for the students. The database has also been provided in the tutor resources folder.

### **Resources**

- Tutor Berryhill database
- Tutor Activity 17 – Wallsherpool Charts database
- Tutor Creating Multi Series Column Charts
- Student Activity 17 – Wallsherpool Charts and Database

		<p><b>Activity 18: Pie Charts</b> Tutor to demonstrate/discuss pie charts. Tutor could use Berryhill school scenario task 8 and Berryhill database for this. For example:</p> <ul style="list-style-type: none"> <li>• query to generate data for the chart including <ul style="list-style-type: none"> <li>– concatenated forename and surname</li> <li>– %paid SUM payment amount / 400, formatted as percent with 2 decimal places</li> </ul> </li> <li>• create report using 'report design' option</li> <li>• using the 'insert chart' option</li> <li>• selecting the pie chart type</li> <li>• specifying a query as a data source</li> <li>• chart setting options</li> <li>• legend required <ul style="list-style-type: none"> <li>– format <ul style="list-style-type: none"> <li>○ display data label</li> </ul> </li> </ul> </li> <li>• chart property sheet <ul style="list-style-type: none"> <li>– legend required, suitable font size</li> <li>– chart title – suitable title. Larger font than rest of chart</li> <li>– primary value axis format (change to percentage)</li> </ul> </li> <li>• changing page layout of report to landscape</li> <li>• ensuring report uses full size of page</li> </ul> <p>Note the chart has been included in the Berryhill database (rptChart3).</p> <p>Students to work individually to complete Activity 18 – Wallsherpool Charts. The database has been provided for the students. The database has also been provided in the tutor resources folder.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Berryhill database</li> <li>• Tutor Activity 18 – Wallsherpool Charts database</li> <li>• Tutor Creating Pie Charts</li> <li>• Student Activity 18 – Wallsherpool Charts and Database</li> </ul>	
12	<p><b>19.2 User interface</b></p> <p>19.2.5 Be able to evaluate the appropriateness and effectiveness of a user interface and justify features in relation to a given problem.</p>	<p><b>Activity 19: User interface design</b> Tutor to split students into pairs/small groups to research usability and user interface design for forms.</p> <p>Pairs/groups to discuss research findings and write a list of what they think are the most important considerations.</p> <p>Pairs/groups to report to whole class.</p>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Adaptive learning</li> <li>• Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>

There are no right or wrong answers. Activities that follow will hone understanding and appreciation of this topic.

**Resources**

- Tutor and Student <https://www.usability.gov/what-and-why/user-interface-design.html>
- Tutor and Student <https://www.ventureharbour.com/form-design-best-practices/>

**Activity 20: Assessing a user interface**

Tutor to explain the keyword 'assess' in terms of the taxonomy for the unit:

- give careful consideration to all the factors or events that apply and identify which are the most important or relevant.
- make judgements about significance/importance.
- do not recommend improvements

Tutor to use Activity20\_SampleAssessment.pdf to show an approach to answering these questions i.e.:

- list the factors that are of importance
- assess whether the form takes each factor into account and judge the significance

Tutor to split students into pairs to evaluate appropriateness and effectiveness of a user interface.

Pairs to complete Activity 20 – Assessing Interface. The database has been provided for the students. The database has also been provided in the tutor resources folder.

**Resources**

- Tutor Activity20\_SampleAssessment.pdf
- Student Activity20 – Assessing Interface
- Tutor and Student <https://www.usability.gov/what-and-why/user-interface-design.html>
- Tutor and Student <https://www.ventureharbour.com/form-design-best-practices/>

		<p><b>Activity 21: Analyse a user interface and make recommendations</b>  Tutor to explain the keywords 'analyse' and 'recommend' in terms of the taxonomy for this unit:</p> <ul style="list-style-type: none"> <li>• examine elements in detail (analyse)</li> <li>• provide support for a course of action. (recommend)</li> </ul> <p>Tutor to use Activity21_SampleAnalysis.pdf to show an approach to analysing and making recommendations for improvements.</p> <p>Tutor to split students into pairs to analyse a user interface and make recommendations for improvement.</p> <p>Pairs to complete Activity 21 – Analyse Interface. Students should use the Activity20-User_Interface-Evaluation database in this activity.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Activity21_SampleAssessment.pdf</li> <li>• Student Activity21 – Assessing Interface</li> <li>• Tutor and Student <a href="https://www.usability.gov/what-and-why/user-interface-design.html">https://www.usability.gov/what-and-why/user-interface-design.html</a></li> <li>• Tutor and Student <a href="https://www.ventureharbour.com/form-design-best-practices/">https://www.ventureharbour.com/form-design-best-practices/</a></li> </ul>	
13-15	<p><b>18.3 Data entry and validation techniques</b></p> <p>18.3.4 Be able to use techniques to aid and improve the quality of data entry:</p> <ol style="list-style-type: none"> <li>dropdown lists</li> <li>radio buttons</li> <li>check boxes</li> <li>automated processes</li> </ol> <p><b>19.2 User interface</b></p> <p>19.2.1 be able to create:</p> <ol style="list-style-type: none"> <li>data entry forms to enter new data</li> <li>action buttons</li> </ol>	<p><b>Activity 22: Creating and customising a data entry form to enter new data</b>  Tutor to demonstrate/discuss columnar forms for data entry. Tutor could use Berryhill school scenario task 1 and Berryhill database for this. For example:</p> <ul style="list-style-type: none"> <li>• using the form wizard to create a basic form based upon tblStudent</li> <li>• getting the form ready for data entry (form property sheet, data, data entry – Yes)</li> <li>• removing form header background colour</li> <li>• suitable title</li> <li>• suitable instructions for use</li> <li>• suitable labels</li> <li>• suitable field widths</li> <li>• suitable layout</li> <li>• suitable tab order</li> <li>• changing a field to a value list combo box (PaymentInstalments, RelationshipToStudent) <ul style="list-style-type: none"> <li>○ right-click field, select Change To, select Combo Box</li> <li>○ go to field Property Sheet, select Data tab</li> <li>○ set Row Source Type to Value List combo box</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Adaptive learning</li> <li>• Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>

<p>19.2.3 Be able to create system outputs and user interface components that support users through:</p> <ol style="list-style-type: none"> <li>selecting/omitting fields</li> <li>appropriate help text</li> <li>meaningful error messages</li> <li>sensible and meaningful data entry field names</li> <li>data entry features       <ol style="list-style-type: none"> <li>dropdown</li> <li>radio buttons</li> <li>input masks</li> </ol> </li> <li>Restricting layout</li> <li>intuitive layout</li> <li>features of good design</li> </ol> <p>19.2.4 Be able to create and use macros to add functionality to user interface objects and/or shorten user processes, including:</p> <ol style="list-style-type: none"> <li>use of program flow to add conditional response (if, then, else)</li> <li>filter data/perform queries</li> </ol>	<ul style="list-style-type: none"> <li>○ set Row Source to the values: 1;2;3;4 (PaymentInstalments) or "Mother";"Father";"Grandmother";"Grandfather";"Guardian" (RelationshipToStudent)</li> <li>○ set Limit To List to Yes</li> <li>○ set Allow Value List Edits to No</li> </ul> <ul style="list-style-type: none"> <li>● removing navigation buttons (form Property Sheet, Format, Navigation Buttons – No)</li> <li>● removing record selectors (form Property Sheet, Format, Record Selectors – No)</li> <li>● removing scroll bars (form Property Sheet, Format, Scroll Bars – neither)</li> <li>● adding a form caption (form Property Sheet, Format, Caption)</li> <li>● determining fields that should be automatically generated (StudentID)       <ul style="list-style-type: none"> <li>– disabling (field property sheet, data, enabled – No)</li> </ul> </li> </ul> <p>Note the form has been included in the database (frmAddStudent1).</p> <p>Students to work individually to complete Activity 22 – Wallsherpool Student Form. The database has been provided for the students (Wallsherpool_Forms). The database has also been provided in the tutor resources folder.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>● Tutor Berryhill database and Wallsherpool_Forms database</li> <li>● Student Activity22-Wallsherpool Student Form and Wallsherpool Forms database</li> <li>● <a href="https://www.webucator.com/how-to/how-create-form-with-the-form-wizard-microsoft-access.cfm">https://www.webucator.com/how-to/how-create-form-with-the-form-wizard-microsoft-access.cfm</a> (creating a form using the form wizard)</li> <li>● <a href="https://www.quackit.com/microsoft_access/microsoft_access_2016/how-to/how_to_change_a_form_header_background_color_in_access_2016.cfm">https://www.quackit.com/microsoft_access/microsoft_access_2016/how-to/how_to_change_a_form_header_background_color_in_access_2016.cfm</a> (removing background colour of form header)</li> <li>● <a href="https://www.brainbell.com/tutorials/ms-office/Access_2003/Aligning_Controls_With_One_Another.htm">https://www.brainbell.com/tutorials/ms-office/Access_2003/Aligning_Controls_With_One_Another.htm</a> (aligning controls)</li> <li>● <a href="https://www.techrepublic.com/blog/microsoft-office/prevent-users-from-changing-data-by-using-access-form-control-properties/">https://www.techrepublic.com/blog/microsoft-office/prevent-users-from-changing-data-by-using-access-form-control-properties/</a> (disabling fields)</li> <li>● <a href="https://www.youtube.com/watch?v=0A1N-HbKuR8">https://www.youtube.com/watch?v=0A1N-HbKuR8</a> (setting tab order)</li> <li>● <a href="https://support.office.com/en-us/article/add-a-list-box-or-combo-box-70abf4a9-0439-4885-9099-b9fa83517603">https://support.office.com/en-us/article/add-a-list-box-or-combo-box-70abf4a9-0439-4885-9099-b9fa83517603</a> (combo boxes)</li> </ul>	
--	---	--

**Activity 23: Automating a data entry form to enter new data**

Tutor to demonstrate/discuss automation. Tutor could use Berryhill school scenario task 1 and Berryhill database for this. For example, the StudentID needs to be generated and the save process automated:

- generating the StudentID (only required if AutoNumber was not used)
- automating the save process
  - Code example 1
    - checking data entry is present and displaying error message if it is not
    - saving the data
    - displaying a save message
    - clearing the form ready for next data entry
    - Note the form has been included in the database (frmAddStudent1\_Automated)
  - Macro example 1
    - save macro created using the wizard (relies on validation being present in the underlying table)
    - adding a save message
    - clearing form ready for next data entry
    - Note the form has been included in the database (frmAddStudent1\_Automated\_Macro\_1)
  - Macro example 2
    - checking data entry is present and displaying error message if it is not
    - saving the data
    - displaying a save message
    - clearing the form ready for next data entry
    - Note the form has been included in the database (frmAddStudent1\_Automated\_Macro\_2)

Students to work individually to complete Activity 23–Wallsherpool Student Form. They should use the database they used in Activity 22. This database has also been provided in the tutor resources folder.

**Resources**

- Tutor Berryhill database and Wallsherpool\_Forms database
- Student Activity23-Wallsherpool Student Form

		<p><b>Activity 24: Creating and Automating a data entry form to enter new data</b>  Students to work individually to complete Activity 24–Wallsherpool Production Form. They should use the database they used in Activity 23. This database has also been provided in the tutor resources folder.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Wallsherpool_Forms database</li> <li>• Student Activity24-Wallsherpool Student Form</li> </ul>	
16-18	<p><b>18.3 Data entry and validation techniques</b></p> <p>18.3.4 Be able to use techniques to aid and improve the quality of data entry:</p> <ol style="list-style-type: none"> <li>dropdown lists</li> <li>radio buttons</li> <li>check boxes</li> <li>automated processes</li> </ol> <p><b>19.2 User interface</b></p> <p>19.2.1 be able to create:</p> <ol style="list-style-type: none"> <li>data entry forms to amend existing data</li> <li>action buttons</li> </ol> <p>19.2.3 Be able to create system outputs and user interface components that support users through:</p> <ol style="list-style-type: none"> <li>selecting/omitting fields</li> <li>appropriate help text</li> <li>meaningful error messages</li> <li>sensible and meaningful data entry field names</li> <li>data entry features</li> <li>dropdown</li> </ol>	<p><b>Activity 25: Creating and customising a data entry form to amend existing data</b>  Tutor to demonstrate/discuss difference between data entry form and a form to amend existing data. For example:</p> <ul style="list-style-type: none"> <li>• should allow the user to easily find the record to edit</li> <li>• should prompt the user to save changes if they try to leave without doing so</li> <li>• should ensure only valid data is acceptable.</li> </ul> <p>Tutor could use the Berryhill database for this. For example:</p> <ul style="list-style-type: none"> <li>• copy and paste frmStudent1_Automated and name it fromEditStudent_Automated</li> <li>• set 'Data Entry' to No so that the form does not open with a blank record</li> <li>• set 'Cycle' to 'Current Record' so that the user cannot move to another record when using the tab key</li> <li>• amend the title and labels on the form to reflect the form's purpose i.e. editing</li> <li>• add a combo box on the form to find a record</li> <li>• change the caption on the save button to 'Save Changes'</li> <li>• add code to check whether the form is 'Dirty' (changes have been made)</li> <li>• remove the code that takes the user to a new record after the save has taken place</li> <li>• edit the save message to say the edit has been saved</li> </ul> <p>Students to work individually to complete Activity 25–Wallsherpool Edit Student Form. They should use the database they used in Activity 24.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Berryhill database and Wallsherpool_Forms database</li> <li>• Tutor CreatingAnEditForm-SingleComboBox</li> <li>• Student Activity25-Wallsherpool Edit Student Form</li> </ul>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Adaptive learning</li> <li>• Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>

	<p>a. radio buttons b. input masks c. Restricting layout d. intuitive layout e. features of good design</p> <p>19.2.4 Be able to create and use macros to add functionality to user interface objects and/or shorten user processes, including:</p> <p>a. use of program flow to add conditional response (if, then, else) b. filter data/perform queries</p>	<p><b>Activity 26: Creating and customising a data entry form to amend existing data</b> Students to work individually to complete Activity 26–Wallsherpool Edit Production Form. They should use the database they used in Activity 25. This database has also been provided in the tutor resources folder.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Wallsherpool_Forms database</li> <li>• Tutor CreatingAnEditForm-SingleComboBox</li> <li>• Student Activity26-Wallsherpool Edit Production Form</li> </ul>	
		<p><b>Activity 27: Creating and customising a data entry form to amend existing data</b> Students to work individually to complete Activity 27–Wallsherpool Edit Character Form. They should use the database they used in Activity 26. This database has also been provided in the tutor resources folder.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Wallsherpool_Forms database</li> <li>• Tutor CreatingAnEditForm-TwoComboBoxes</li> <li>• Student Activity27-Wallsherpool Edit Character Form</li> </ul>	
19-20	<p><b>19.2 User interface</b></p> <p>19.2.1 be able to create:</p> <p>a. menus/dashboards b. action buttons</p> <p>19.2.4 Be able to create and use macros to add functionality to user interface objects and/or shorten user processes, including:</p> <p>a. use of program flow to add conditional response (if, then, else) b. data import/export c. filter data/perform queries d. user interface management (e.g. open form, close menu)</p>	<p><b>Activity 28: Menus</b> Tutor to demonstrate/discuss menus. For example:</p> <ul style="list-style-type: none"> <li>• meaningful menu title</li> <li>• buttons to carry out actions. For example: <ul style="list-style-type: none"> <li>– open form</li> <li>– open report</li> <li>– open query</li> </ul> </li> <li>• radio buttons to carry out actions. For example, opening relevant form when option is selected</li> <li>• checking whether a user wants to print or preview a report (message box and If statement)</li> <li>• checking whether forms, queries should open. (If statements and DCount)</li> </ul> <p>Tutor could use the Berrymill database for this.</p> <p>Students to work individually to complete Activity 28–Wallsherpool Menu. They should use the database they used in Activity 27. This database has also been provided in the tutor resources folder.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• Tutor Berrymill database and Wallsherpool_Forms database</li> <li>• Student Activity28-Wallsherpool Menu</li> </ul>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Adaptive learning</li> <li>• Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>

<p><b>19.1 Using database software</b></p> <p>19.1.6 Be able to import data from external sources:</p> <ol style="list-style-type: none"> <li>.xls, .xlsx</li> <li>.csv</li> <li>.txt</li> </ol> <p>19.1.7 Be able to export data to external sources:</p> <ol style="list-style-type: none"> <li>.xls, .xlsx</li> <li>.csv</li> <li>.txt</li> <li>.html</li> <li>.pdf</li> </ol>	<p><b>Activity 29: Import/export</b></p> <p>Tutor to demonstrate/discuss different file types when importing/exporting including:</p> <ul style="list-style-type: none"> <li>import <ul style="list-style-type: none"> <li>.xls, .xlsx</li> <li>.csv</li> <li>.txt (already covered)</li> </ul> </li> <li>export <ul style="list-style-type: none"> <li>.xls, .xlsx</li> <li>.csv</li> <li>.txt</li> <li>.html</li> <li>.pdf (already covered)</li> </ul> </li> </ul> <p>Tutor may use the Berrymill database for this. Objects within the database can be exported in the formats given. Data can then be imported from these exports in the formats given.</p> <p>Students to work individually to export of data from and import data to their Wallsherpool database (Activity 28) using the formats given.</p> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>Tutor Berrymill database</li> <li><a href="https://blog.ip2location.com/knowledge-base/how-to-import-csv-into-microsoft-access-database/">https://blog.ip2location.com/knowledge-base/how-to-import-csv-into-microsoft-access-database/</a> (importing .csv)</li> <li><a href="https://smallbusiness.chron.com/convert-excel-documents-microsoft-access-40490.html">https://smallbusiness.chron.com/convert-excel-documents-microsoft-access-40490.html</a> (importing Excel files)</li> <li><a href="https://support.office.com/en-us/article/export-data-to-excel-64e974e6-ae43-4301-a53e-20463655b1a9">https://support.office.com/en-us/article/export-data-to-excel-64e974e6-ae43-4301-a53e-20463655b1a9</a> (export as Excel)</li> <li><a href="https://www.techwalla.com/articles/how-to-export-access-query-as-csv">https://www.techwalla.com/articles/how-to-export-access-query-as-csv</a> (export as .csv)</li> <li><a href="https://www.techrepublic.com/article/turn-access-forms-into-html-pages-in-five-steps/">https://www.techrepublic.com/article/turn-access-forms-into-html-pages-in-five-steps/</a> (export as html)</li> </ul>	
<p>19.2.4 Be able to create and use macros to add functionality to user interface objects and/or shorten user processes, including: data export</p>	<p><b>Activity 30: Automate import/export of objects</b></p> <p>Tutor to discuss/demonstrate using embedded macros to automate the export of objects in these formats:</p> <ul style="list-style-type: none"> <li>xls, .xlsx</li> <li>.txt</li> <li>.html</li> <li>.pdf</li> </ul>	

		<p>Tutor may use the Berryhill database for this. Objects within the database can be exported using embedded macros.</p> <p>Students to work individually to automate the export of data their Wallsherpool database (Activity 28) using the formats given.</p> <p><b>Resources</b> Tutor Export_Using_Macro</p>	
21-23	<p><b>19.3 Testing the solution</b></p> <p>19.3.1 Understand the need to test that solutions to a problem work as intended and are fit for purpose.</p> <p>19.3.2 Understand the importance of testing as a database solution is developed, with intended users.</p> <p>19.3.3 Be able to design and carry out tests to ensure that their solution meets the specified requirements such as:</p> <ol style="list-style-type: none"> <li>menus work correctly</li> <li>validation checks prevent unacceptable data from being entered</li> <li>outputs are complete, accurate and in the required format.</li> </ol>	<p><b>Activity 31: Testing</b></p> <p>Tutor to discuss testing including:</p> <ul style="list-style-type: none"> <li>• why it is important to test solutions</li> <li>• why it is important to test a solution, as it is being developed, with users</li> <li>• functional testing to test the solutions work as intended</li> <li>• usability testing to test that solutions are fit for purpose</li> <li>• types of functional tests <ul style="list-style-type: none"> <li>– valid test data – data/inputs that the software product should accept</li> <li>– invalid – data/inputs that the software product should not accept</li> <li>– valid extreme - data/inputs that are valid but at the boundary of acceptable</li> <li>– invalid extreme – data/inputs that are valid but just outside the acceptable boundary</li> <li>– erroneous – data/inputs that are of the wrong data type.</li> </ul> </li> </ul> <p>Students to work in pairs or small groups to answer these questions:</p> <ol style="list-style-type: none"> <li>1. Describe the difference between functional and usability testing</li> <li>2. Give one potential consequence of not carrying out functional testing</li> <li>3. Give one potential consequence of not carrying out usability testing</li> <li>4. A database table includes validation on a number field named Quantity. The Quantity must be at least 1 and below 5. Give example test data for each type of functional test.</li> </ol> <p><b>Resources</b></p> <ul style="list-style-type: none"> <li>• <a href="https://www.testdevlab.com/blog/2018/07/importance-of-software-testing/">https://www.testdevlab.com/blog/2018/07/importance-of-software-testing/</a> (why testing is needed)</li> <li>• <a href="https://www.ibm.com/topics/software-testing">https://www.ibm.com/topics/software-testing</a> (functional and usability testing overview)</li> <li>• <a href="https://www.usability.gov/how-to-and-tools/methods/planning-usability-testing.html">https://www.usability.gov/how-to-and-tools/methods/planning-usability-testing.html</a> (planning usability testing)</li> </ul>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Adaptive learning</li> <li>• Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>

**Activity 32: Test plan**

Tutor to discuss and demonstrate create a test plan to carry out functional testing of a table:

- Test number
- Purpose of test
- Type of test (valid, invalid, valid extreme, invalid extreme, erroneous)
- Test data (specific for all fields in the record)
- Expected results (specific)

Tutor could use the Berry mill database and the testing template to do this. Students to work individually to create a test plan for the character table in the Wallsherpool database used in activity 28.

**Resources**

- Tutor Berry mill database
- Tutor Berry mill\_testing\_template\_example (note actual results should not be discussed until Activity 31).
- Tutor testing\_template
- Student Activity32\_Testing\_template

**Activity 33: Test results**

Tutor to discuss and demonstrate documenting the results of table testing:

- screenprint showing **all** of the test data in situ and the result e.g. error message
- comments if test did not produce expected results
- details of how the error was amended
- screenprint of retest results.

Tutor could use the Berry mill database and the testing template to do this.

Students to swap the test plan they created in Activity 30 with another student and carry out table testing using that plan. Students to discuss the results of the testing with each other.

**Resources**

- Tutor Berry mill database
- Tutor Berry mill\_testing\_template\_example (note actual results should not be discussed until Activity 31).
- Tutor testing\_template

		<p><b>Activity 32: Testing input forms</b> Students to work individually to create a test plan and test the form created to add a student in their Wallsherpool database (Activity 28).</p> <p><b>Resources</b> Student Activity32_Testing_template</p> <hr/> <p><b>Activity 33: Testing edit forms</b> Students to work individually to create a test plan and test the form created to edit a character in their Wallsherpool database (Activity 28).</p> <p><b>Resources</b> Student Activity32_Testing_template</p> <hr/> <p><b>Activity 34: Testing menus</b> Students to work individually to create a test plan and test the menu created to add a student in their Wallsherpool database (Activity 28).</p> <p><b>Resources</b> Student Activity32_Testing_template</p>	
24-26	<p><b>18.1 Structuring data</b> 18.1.4 Be able to evaluate the appropriateness and effectiveness of a data structure in relation to the requirements of a given scenario</p> <p><b>18.2 Relational data structures</b> 18.2.3 Be able to evaluate the appropriateness and effectiveness of a relational data structure in relation to the requirements of a given scenario</p> <p><b>18.3 Data entry and validation techniques</b> 18.3.5 Be able to evaluate the use of data accuracy techniques in relation to a given problem</p>	<p><b>Activity 35: Evaluating structuring data and relational data structures</b> Tutor to remind students that there are two different approaches to evaluating and the differences between the two.</p> <ul style="list-style-type: none"> <li>• analyse and recommend</li> <li>• assess</li> </ul> <p>Tutor to explain that evaluating the structure of data (18.1) is different to evaluating relational data structures (18.2) though they would tend to be evaluated together. This is an opportunity for students to demonstrate their knowledge of 18.1 and 18.2 by determining how appropriate and effective a data structure and relational data structure are in terms of a given scenario.</p> <p>The tutor should provide database examples that can be evaluated through analysis and recommendations or assessment. The databases should include flaws. For example:</p> <ul style="list-style-type: none"> <li>• incorrect data types</li> <li>• incorrect formats/input masks etc for the data present</li> <li>• tables based on data that has not been fully normalised</li> </ul>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Adaptive learning</li> <li>• Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>

	<p><b>19.1 Using database software</b></p> <p>19.1.8 Be able to evaluate the effectiveness and appropriateness of selected database tools and techniques used in relation to a given problem</p>	<p><b>Activity 36: Evaluating data entry, validation, tools and techniques</b></p> <p>Tutor to remind students that there are two different approaches to evaluating and the differences between the two.</p> <ul style="list-style-type: none"> <li>• analyse and recommend</li> <li>• assess</li> </ul> <p>Tutor to explain that 18.3 and 19.1 would usually be incorporated into one evaluation.</p> <p>Students should work individually to assess data entry and validation techniques used in their own final Wallsherpool database.</p> <p>Student should then work in pairs to evaluate each other's final Wallsherpool database by analysing the database in terms of data entry and validation techniques and providing recommendations for improvements.</p>	
27-28	All	<p><b>Mock exam and feedback</b></p> <p>Students to complete the sample assessment material. Student must ensure that screenprints are legible and that they fully show the methods they have used e.g. queries used for reports, code for forms etc.</p> <p>Tutor to mark using the mark scheme and provide feedback</p>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Adaptive learning</li> <li>• Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>
29-30	All	<p><b>Mock exam and feedback</b></p> <p>Students to complete the additional sample assessment material</p> <p>Tutor to mark using the mark scheme and provide feedback</p>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Adaptive learning</li> <li>• Adapting prior knowledge, skills and experience of IT to deal with new situations/contexts</li> </ul>

FOR INFORMATION ABOUT EDEXCEL, BTEC OR LCCI QUALIFICATIONS VISIT [QUALIFICATIONS.PEARSON.COM](http://QUALIFICATIONS.PEARSON.COM)  
EDEXCEL IS A REGISTERED TRADEMARK OF PEARSON EDUCATION LIMITED PEARSON EDUCATION LIMITED.  
REGISTERED IN ENGLAND AND WALES NO. 872828  
REGISTERED OFFICE: 80 STRAND, LONDON WC2R 0RL  
VAT REG NO GB 278 537121

GETTY IMAGES: ALEX BELMONLINSKY

