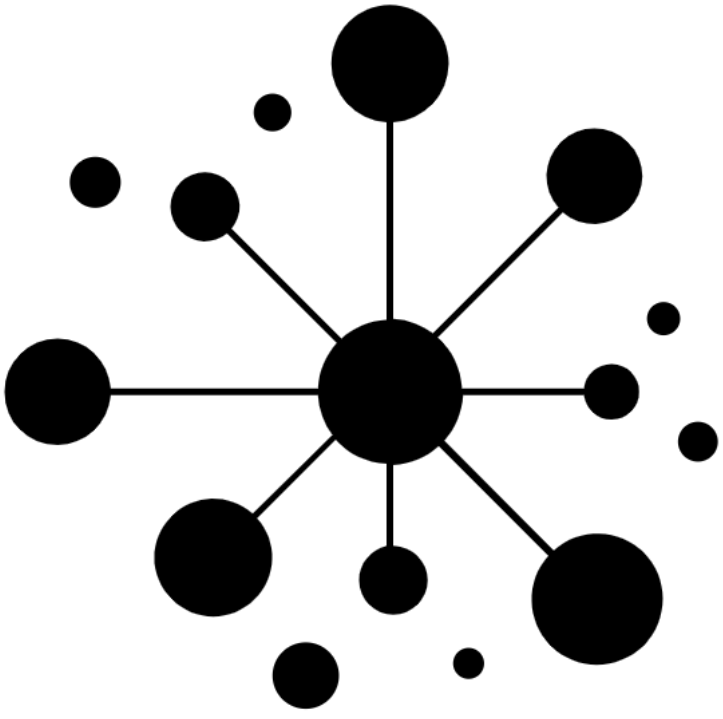
Department of Education

GOVERNMENT OF WESTERN AUSTRALIA

Year 12 Computer Science ATAR
Managing Data



Department of Education Western Australia 2020

Computer Science ATAR Year 12

Resource Package

Managing Data

Notes

Instructions to Students

This resource package provides students with learning materials for the Computer Science ATAR Year 12 course. The package focuses on content from the Managing Data content area in the 2020 Year 12 Syllabus. The content covered from the syllabus is listed on Page 2 of the Notes document.

This package is designed to support the program students are completing at their school. If feedback is required when completing this package, students should consult their teacher.

This resource package consists of two parts:

- The **Notes** document provides an explanation of syllabus content concepts. This section is designed to develop the knowledge component of the syllabus.
  The Notes document is in PDF form.
- The **Activities** document provides an opportunity for students to actively engage with Computer Science content. The activities are designed to develop both the knowledge and the skills components of the syllabus.
  The Activities document is provided in both PDF and Microsoft Word format. Students can print the PDF version or work directly on the editable Microsoft Word document (see attachments in this PDF).

Students should read one section at a time of the **Notes** document. The **Activities,** indicated in blue text at the end of the section, should be completed before moving onto the next Notes section. If students are completing the Activities in Word, it is recommended that they open both the Notes and the Activities documents and switch between reading the content and completing the associated Activities.

It is recommended that students further investigate concepts covered in this resource package by conducting their own research using the internet. There are many resources, for example on YouTube, that will further develop understandings of the Computer Science concepts.

Syllabus Content

Topic	Element	Student Check ✓
databases	types of physical storage of databases	
	online	
	local	
	types of databases	
	distributed	
	centralised	
data warehouses and data marts	structure of data warehouses and data marts	
	role of data mining	
	compare data warehouses and data marts as methods of data storage and distribution	
	ethical implications of the use of data warehouses, data marts and data mining	
data dictionary	purpose of a data dictionary	
	elements of a data dictionary, including: element name, data type, size/format default, description, constraint	
database management	database management system concepts, including:	
	data definition	
	data duplication	
	data integrity, including: referential integrity,	
	domain integrity and entity integrity	
	data redundancy	
	data anomalies, including: insert, delete and update	
	data manipulation	
	data security	
normalisation	normalisation of data to 3rd normal form (NF)	
relational databases	data types	
	relations	
	primary, composite and foreign keys	
	referential integrity	
	relationships, including: set cascade inserts, updates and deletes	
	cardinality (1:1, 1:M, M:1, M:N)	
	validation rules	
	forms	
	reports	
	simple queries using SQL (up to two tables), including insert, update and select queries	
	queries across multiple tables using appropriate database tools, including the following:	
	parameter	
	calculated field	
	concatenated field	
	aggregation	
	update	

	delete	
	make table	
Open systems	role of open systems in:	
	database interconnectivity	
	database development	
	database management	
	data driven websites	
ER diagrams	data modelling using Chen's notation entity relationship (ER) diagrams	
database documentation	purpose of database documentation for the user	
user interfaces	design considerations for visual interfaces and navigation systems within database systems, including:	
	readability	
	navigation	
	logical order	
	inclusivity	

Introduction

Data

Data is raw facts that represent real-world things. For example, a student's name or date of birth is data. Data can be text, numeric, graphical or audio.

Information

Information is data that is organised in a meaningful manner and provides value beyond the individual facts or items of data. For example, the date ***14/12/2020*** is data whereas "The date the assignment is due is 14/12/2005" is information.

A database is a **collection of related data** organised using records and tables. Databases enable data to be stored and managed to readily provide information.

Single Table Databases

Microsoft Excel is an example of an application that provides single table database functions. A single table database organises records into one table or file that cannot be cross-referenced with data in any other file or table. For this reason, single table databases such as Microsoft Excel are only suitable for relatively simple collections of data. When complex data is stored in a single table database a number of problems, called anomalies, are likely to occur. These database anomalies are covered later in the course.

The following table shows data for a human resource company stored in a single Microsoft Excel worksheet. There are problems that result from this table that can only be resolved through the use of a **relational database**.

Staff ID	Staff Surname	Staff First Name	Branch Name	Branch Phone
1	Jones	Nancy	South Perth Office	95449999
2	Fuller	Andrew	Como Office	93337444
3	Kent	Janice	South Perth Office	95449999
4	Sarraf	Ghina	Bunbury Office	96456782
5	Nguyen	Robert	South Perth Office	95449999
6	King	Jonathan	Como Office	93337444
8	Morris	Linda	South Perth Office	95449999

Complete Activity One

Relational Databases

Data in a relational database is stored in **multiple tables (relations)** that are interconnected by links called **relationships**. This overcomes many of the problems that occur in single table databases such as Microsoft Excel. Relational databases are implemented using a **database management system (DBMS)**. Microsoft Access, FileMaker Pro and Oracle are three examples of database management systems.

Below is an example of a simple relational database that stores the human resource company information in two linked table: Staff and Branch.

Staff

Staff ID	Staff Surname	Staff First Name	fk Branch ID
1	Jones	Nancy	1
2	Fuller	Andrew	2
3	Kent	Janice	1
4	Sarraf	Ghina	3
5	Nguyen	Robert	1
6	King	Jonathan	2
7	Morris	Linda	1

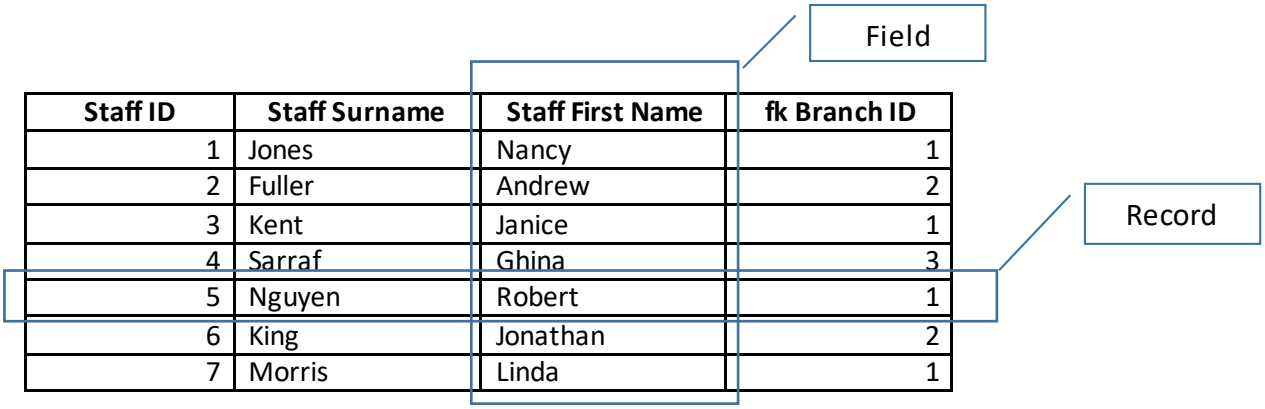
Branch

Branch ID	Branch Name	Branch Phone
1	South Perth Office	95449999
2	Como Office	93337444
3	Bunbury Office	96456782

Using multiple tables in a DBMS allows data to be stored without repetition and improves the accuracy of the database. In a DBMS, tables are stored in a single file, for example a .accdb file when using Microsoft Access, or stored across multiple files, for example when using Oracle.

Table/Relation

A table is the basic structure that stores data about a particular subject. Tables are made up of rows called **records** and columns called **fields**.



Staff ID	Staff Surname	Staff First Name	fk Branch ID
1	Jones	Nancy	1
2	Fuller	Andrew	2
3	Kent	Janice	1
4	Sarraf	Ghina	3
5	Nguyen	Robert	1
6	King	Jonathan	2
7	Morris	Linda	1

In relational database theory, the table structure is also known as a **relation** and consists of tuples (rows) and attributes (columns).

Entity

Each table stores information about a specific subject called an **entity**. For example, the entity represented by the table above is a Staff. Entities can be anything distinguishable that needs to be recorded such as people, objects, places or events. When planning the tables in a database, an entity relationship diagram can be used. This will be covered later in the course.

Records

A record contains information about one instance of an entity. In the example above, each record holds information about one staff member. In a table, a record is a single row.

Fields

A field is an attribute or characteristic of an entity. Field names are the headings for each column. Examples from the Staff table are Surname and First Name.

Primary Key

Each table in a relational database must have a **primary key**. The primary key is used to uniquely identify each record in the table. There cannot be any repeated values in a primary key field; each value in the primary key field must be unique. In the table above, the primary key is Staff ID.

The primary key may be a single field or it may be a combination of fields known as a **composite** primary key. A combination of fields is used where there is no single field that will provide a unique value for a record. For example, a doctor appointment table would need four fields to uniquely identify the record for an appointment: Doctor ID, Patient ID, Appointment Date and Appointment Time.

Often it is better to create a new field to provide a unique identifier rather than use a composite key. For example, it may be easier to create field called Appointment ID to use as the primary key for the appointment table.

Complete Activity Two

Relationships

Relational databases have multiple tables. **Relationships** are used to link the data held in each table. For example, suppose an organisation wants to create a database to record information about the people who book to attend different events. Three tables would be needed, People, Events and Bookings and a relationship would need to be defined between each table to link a person to a booking and a booking to an event.

The tables required to store this information would be:

People - a table containing details about each person.

People ID	Title	Surname	First Name	Address	City	PostCode	Phone	Date Of Birth	Married
1	Mrs	Davolio	Theresa	20 Barrack Street	Perth	6000	355 9857	6/12/2000	Yes
2	Mr	Singh	Anmed	90 Capital Way	Applecross	6953	455 9482	19/02/1999	No
3	Miss	Leverling	Janice	72 Moss Bay Rd.	Perth	6000	285 3412	30/08/1997	Yes
4	Ms	Wyatt	Coral	41 Redmond Rd.	Bentley	6102	286 3312	19/09/1993	Yes
5	Mr	Lee	David	14 Garrett Road	Perth	6000	275 4848	4/03/2000	Yes
6	Mr	Suyama	Michael	100 Eric Street	Como	6152	325 6773	2/07/1980	No
7	Mr	Brookes-King	Jonathan	17 Rose Avenue	Perth	6000	365 5598	29/05/1978	Yes
8	Miss	Callahan	Linda	17 Jarrah Street	Willetton	6955	272 1189	9/01/1989	Yes
9	Ms	Dodsworth	Annabella	7 High Rd.	Northbridge	6003	272 4444	27/11/1991	No

Events - a table that contains details about each event.

Event ID	Event Name	Venue	Date	Time	Capacity	Fee
1	Access National Conference	Tuart College	9/11/2019	4:30:00 PM	250	$120.00
2	Apple Special Event	Burswood	11/10/2019	5:30:00 PM	500	$0.00
3	Logitech Product Launch	Perth	10/12/2019	10:00:00 AM	300	$120.00
4	2020 Internet Conference	Northbridge	22/01/2020	4:00:00 PM	45	$300.00
5	Apple Special Event	Hyatt	4/04/2018	8:00:00 PM	400	$0.00
6	Management Seminar	Observation City	10/09/2019	8:00:00 AM	220	$500.00
7	Access Developers Conference	Burswood	5/05/2018	9:00:00 AM	50	$100.00
8	Sales Investment Seminar	Burswood	21/08/2019	9:00:00 AM	100	$75.00
9	Adobe Seminar	Hyatt	22/08/2019	5:00:00 PM	200	$25.00
10	Sales Investment Seminar	Observation City	10/10/2018	12:00:00 PM	150	$80.00

Bookings - a table that contains booking details about who attends each event.

Booking ID	fk People	fk Event ID	Booking Date	Confirmed	Payment Recieved
1	8	1	1/10/2019	Yes	Yes
2	2	2	1/09/2019	Yes	Yes
3	3	2	22/09/2019	Yes	Yes
4	8	2	23/09/2019	Yes	Yes
5	5	3	7/11/2019	Yes	No
6	3	4	22/12/2019	Yes	No
7	7	4	8/01/2020	Yes	Yes
8	7	6	4/08/2019	Yes	Yes
9	8	6	12/08/2019	No	No
10	1	7	1/03/2018	Yes	Yes
11	2	7	14/04/2018	Yes	No
12	2	8	12/07/2019	Yes	No
13	1	9	10/07/2019	Yes	Yes
14	3	10	12/09/2018	Yes	No
15	8	10	11/09/2018	Yes	Yes

Foreign Key

In the tables above, the relationships have been specified by the addition of two extra fields in the Bookings table: **fk Event ID** and **fk People ID**. These fields record the ID of the event that has been booked and the ID of the person who has made the booking.

Extra fields used to create references in this way are called **Foreign Keys.** A foreign key can be defined as a field in a table that forms a link by storing matching values from the primary key of another table.

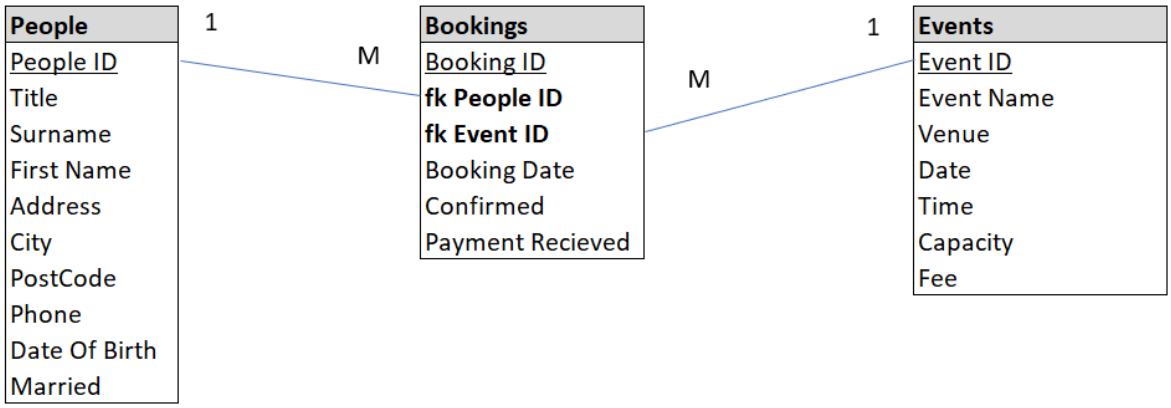
Data Integrity

Data integrity is the accuracy or correctness of data. Data integrity ensures that the data held in a database is as error free as possible and follows the business rules of the organisation. There are three main types of data integrity: referential integrity, entity integrity and domain integrity.

Referential integrity

Referential integrity is a method of ensuring that the data held in foreign keys are kept accurate. Referential integrity ensures that the foreign key only contains values that match the primary key in the linked table.

In the bookings database example above, referential integrity ensures that a booking can only be made for a valid person and a valid event. It also ensures that a person or event record cannot be deleted if there is foreign key value for them in the Bookings table.



It is possible to set different referential actions to manage referential integrity. Two common actions are **cascading update** and **cascading delete**.

Setting cascading update allows the primary key in a relationship to be modified. Any changes to the primary key will automatically update the foreign key in the linked table with the changes. For example, if the Event ID is changed in the Event table, the linked value in the field fk Event ID will also be changed to the same value.

Setting cascading delete allows records in the table at the primary key end of a relationship to be deleted. Matching records in the linked foreign key table will be automatically deleted. For example, if an event is deleted, the booking record and associated people records will be deleted automatically.

Entity integrity

Entity integrity is a method of ensuring that each row in a table is uniquely identified. This ensures that there is a primary key in a table that is unique and never contains null values.

Domain integrity

A domain defines all the possible values that can be entered into a field. Domain integrity specifies what values are allowed in the domain and includes the following rules:

- Each field should have a particular **data type** that ensures data is stored in a consistent type. For example, the data type for Surname in the table above is Text. Other data types include number, date, Boolean (yes/no), currency, or object (such as a photo).
- Each field should have a **field size** that ensures data is stored in the most useful and economical way. For example, the Text data type can be set to a maximum number of characters.
- Each field should have a **default value** that will be automatically created when a new record is added (this is often simply a null value).
- Each field can have **validation rules** that ensure that data entered into the database complies with business rules. This is normally done by creating a restriction on a field to ensure that only valid data is accepted. For example, a validation rule may be set up to ensure that a person's salary can only have numbers entered into the field between $10,000 and $250,000.

Complete Activity Three

Data Redundancy

Data redundancy occurs when the same data is repeated in multiple records in the table. For example, in the table below, the information about the South Perth Office is repeated.

Staff ID	Staff Surname	Staff First Name	Branch Name	Branch Phone
1	Jones	Nancy	South Perth Office	95449999
2	Fuller	Andrew	Como Office	93337444
3	Kent	Janice	South Perth Office	95449999
4	Sarraf	Ghina	Bunbury Office	96456782
5	Nguyen	Robert	South Perth Office	95449999
6	King	Jonathan	Como Office	93337444
8	Morris	Linda	South Perth Office	95449999

Data redundancy causes integrity problems, called **database anomalies**, and should be removed where possible. These database anomalies are described below and the methods for removing or avoiding them are described in the Designing a Database section later in the course.

Database Anomalies

Database anomalies are problems that occur when data is managed in a poorly designed database. This is usually caused because the attributes of more than one entity have been combined in the same table. For example, in the table above two entities, Staff and Branch, exist in the same table.

There are three types of anomalies that can occur: insert, update and delete anomalies.

Update Anomalies

Update anomalies occur when data is stored unnecessarily more than once in a table. Any changes that need to be made to that data will require multiple updates to be carried out. This is inefficient and can cause problems where repeated data is not able to be updated consistently across all records.

In the example table below, if the phone number needs to be changed for the South Perth branch, four records will need be updated rather than just one.

Staff ID	Staff Surname	Staff First Name	Branch Name	Branch Phone
1	Jones	Nancy	**South Perth Office**	**95449999**
2	Fuller	Andrew	Como Office	93337444
3	Kent	Janice	**South Perth Office**	**95449999**
4	Sarraf	Ghina	Bunbury Office	96456782
5	Nguyen	Robert	**South Perth Office**	**95449999**
6	King	Jonathan	Como Office	93337444
8	Morris	Linda	**South Perth Office**	**95449999**

Insert Anomalies

Insert anomalies occur when adding data to a table necessitates the recording of null values in some fields or forces redundant data to be entered that already exists in the table. This is because two or more entities exist in the same table.

In the example table below, if a new office is planned for opening in Albany but there are currently no staff at the office, the details of the office can't be added to the table without creating null values.

Staff ID	Staff Surname	Staff First Name	Branch Name	Branch Phone
1	Jones	Nancy	South Perth Office	95449999
2	Fuller	Andrew	Como Office	93337444
3	Kent	Janice	South Perth Office	95449999
4	Sarraf	Ghina	Bunbury Office	96456782
5	Nguyen	Robert	South Perth Office	95449999
6	King	Jonathan	Como Office	93337444
8	Morris	Linda	South Perth Office	95449999
			Albany Office	**92426222**

Delete Anomalies

Deletion anomalies occur when the deletion of a record containing information about one entity results in the unavoidable loss of fields from another.

In the example table below, if a Ghina Sarraf leaves the organisation and her staff record needs to be deleted, all information about the Bunbury office will also be lost.

Staff ID	Staff Surname	Staff First Name	Branch Name	Branch Phone
1	Jones	Nancy	South Perth Office	95449999
2	Fuller	Andrew	Como Office	93337444
3	Kent	Janice	South Perth Office	95449999
4	**Sarraf**	**Ghina**	**Bunbury Office**	**96456782**
5	Nguyen	Robert	South Perth Office	95449999
6	King	Jonathan	Como Office	93337444
8	Morris	Linda	South Perth Office	95449999

Data Duplication

Data duplication is another issue that can occur in databases. Data duplication occurs where the same data is stored more than once across a database system, for example two tables that contain similar or identical information.

Whereas data redundancy involves the unnecessary repetition of data, data duplication may be necessary in some situations. For example, if a database is being used offsite, an operational copy of the database may be created. The offsite copy would be regularly synchronised with the primary database.

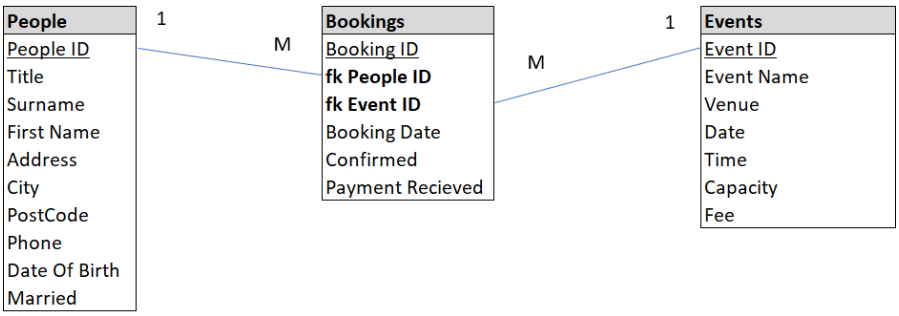
Complete Activity Four

Queries

Queries provide a means for creating new views of the data held in tables. Queries are questions that you ask about the data in a database. For example, "What are the phone numbers of all the people with a surname starting with "M"? You define a query on the fields in one or more tables.

The result of a query will be a set of records that appear like a table. Using queries, the user can create views of the data in the database to find information suited to their specific needs.

In Microsoft Access, for example, queries are created using the "query by example" window. The query below finds all the people whose payment has been received for their booking:



Field:	Firstname	Surname	Booking Date	Payment Received	Event Name	Venue
Table:	People	People	Bookings	Bookings	Events	Events
Sort:						
Show:	☑	☑	☑	☑	☑	☑
Criteria				**Yes**		
Or:						

The set of records displayed by the query is shown below.

First Name	Surname	Booking Date	Payment Recieved	Event Name	Venue
Linda	Callahan	1/10/2019	Yes	Access National Conference	Tuart College
Anmed	Singh	1/09/2019	Yes	Apple Special Event	Burswood
Janice	Leverling	22/09/2019	Yes	Apple Special Event	Burswood
Linda	Callahan	23/09/2019	Yes	Apple Special Event	Burswood
Jonathan	Brookes-King	8/01/2020	Yes	2020 Internet Conference	Northbridge
Jonathan	Brookes-King	4/08/2019	Yes	Management Seminar	Observation City
Theresa	Davolio	1/03/2018	Yes	Access Developers Conference	Burswood
Theresa	Davolio	10/07/2019	Yes	Adobe Seminar	Hyatt
Linda	Callahan	11/09/2018	Yes	Sales Investment Seminar	Observation City

In most database management systems, queries are defined using a language called **Structured Query Language (SQL)**. SQL is written using statements that allow the user to **select** field/s, **from** tables, **where** the data matches given criteria.

The SQL create by Microsoft Access for the above query is:

```
SELECT First Name, Surname, Booking Date, Payment Recieved, Event Name, Venue
FROM People, Bookings, Events
WHERE Bookings.Payment Recieved = Yes
```

The advantage of using a query to find information rather than searching directly in a table (for example using the Find command or a Filter) is that queries are more powerful, multiple queries can be saved for reuse and queries can be used to find information from more than one table at the same time.

Advanced Queries

Advanced queries can be created to ask more complex questions of the database.

- A **concatenate** query joins data from two or more fields together to create a single value. For example, to combine first name and surname in Microsoft Access:

  Staff.[Staff First Name] & " " & Staff.[Staff Surname]

- An **aggregation** query groups identical values from one or more fields and provides calculated data for the group, for example sum, count, average etc. In Microsoft Access this is called a Group By query.
- A **calculated** field can be generated that uses numerical values from fields in a table to carry out a calculation such as addition, subtraction etc. For example, in Microsoft Access a new field called TotalCost could be output from a query to display the result of multiplying the Quantity and Price fields from a table: TotalCost: [Quantity] * [Price).
- A **parameter** query prompts the user to input data when the query is opened. The user's data is used to complete selection criteria that determines what the records will be displayed by the query. This allows the user to easily control the operation of the query each time it is run.

Action Queries

Action queries can be used to create or make changes to data in tables. There are three actions performed by action queries: update, delete and make table.

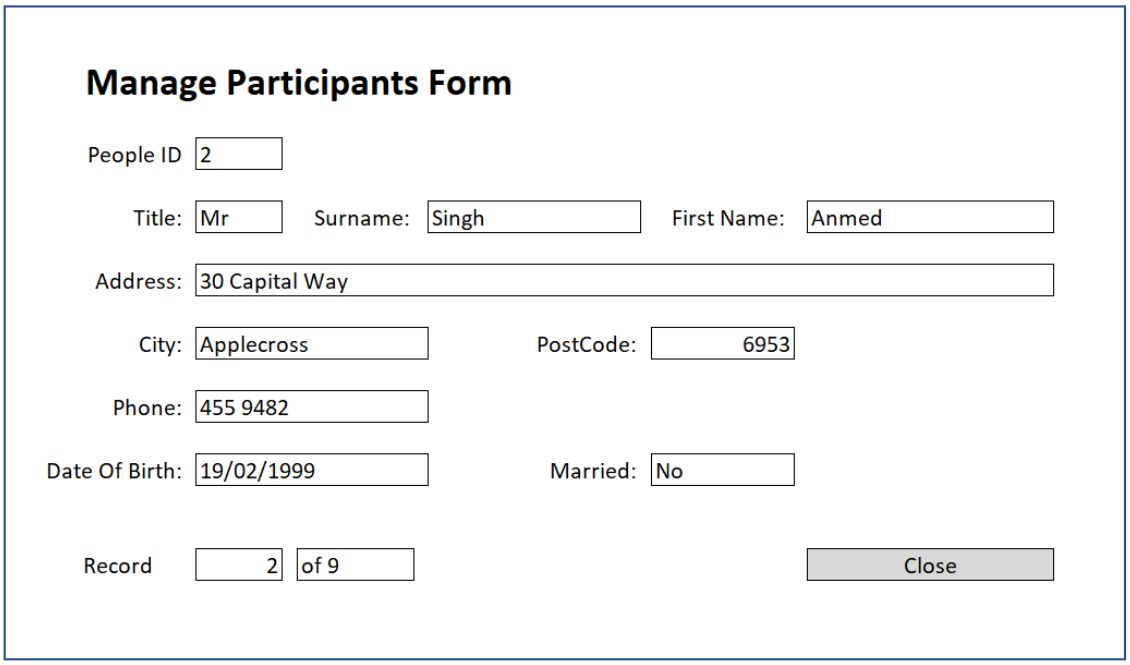
- An **update query** is used to make changes to data in multiple records in an existing table.
- A **delete query** is used to delete entire records from a table. Criteria is used to select which records will be deleted.
- An **insert query** is used to copy data from records in one table to create new records in another table.

Complete Activity Five

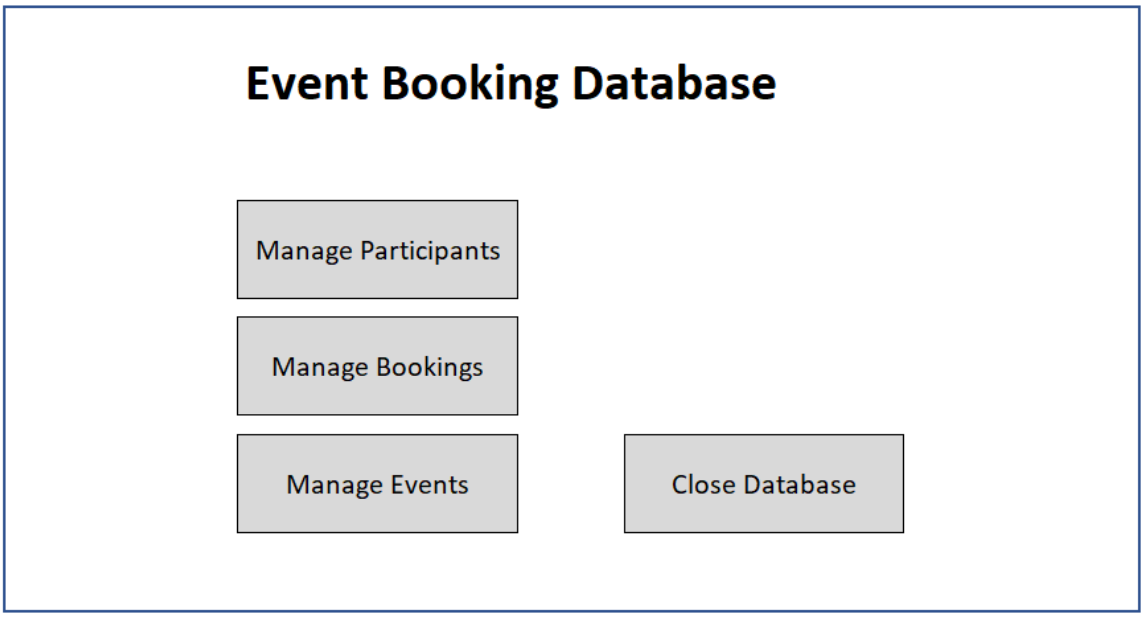
Forms

Forms are used to create a **user-interface** for a database. Forms are commonly used to view data stored in tables and to enter data into tables. Forms can be created inside the database itself or can be created externally to provide access to the data held in a DBMS.

For example, the form below accesses the table People and is used to view existing data, edit data or enter new data into the table.



Forms also enable the user to navigate around the database easily. For example, the form below provides buttons to enable the user to open components of the Event Booking database.



Reports

Reports are used to provide an analysis or summary of the information held in a database. Reports enable a printed copy of information from tables or queries to be created. The user can choose the fields they want to be included and can choose the layout for the information in the report. Reports can be created inside the DBMS, for example in Microsoft Access, or can be created externally to access the data in a DBMS, for example using Crystal Reports.

For example, the report below shows information about event bookings:

Event Bookings Report

Event Name	Venue	Date	Time	First Name	Surname	Phone	Confirmed
2020 Internet Conference	Northbridge	22/01/2020	4:00:00 PM				
				Jonathan	Brookes-King	365 5598	Yes
				Janice	Leverling	285 3412	Yes
Access Developers Conference	Burswood	5/05/2018	9:00:00 AM				
				Anmed	Singh	455 9482	Yes
				Theresa	Davolio	355 9857	Yes
Access National Conference	Tuart College	9/11/2019	4:30:00 PM				
				Linda	Callahan	272 1189	Yes
Adobe Seminar	Hyatt	22/08/2019	5:00:00 PM				
				Theresa	Davolio	355 9857	Yes
Apple Special Event	Burswood	11/10/2019	5:30:00 PM				
				Linda	Callahan	272 1189	Yes
				Janice	Leverling	285 3412	Yes
				Anmed	Singh	455 9482	Yes
Logitech Product Launch	Perth	10/12/2019	10:00:00 AM				
				David	Lee	275 4848	Yes
Management Seminar	Observation City	10/09/2019	8:00:00 AM				
				Linda	Callahan	272 1189	No
				Jonathan	Brookes-King	365 5598	Yes
Sales Investment Seminar	Observation City	10/10/2018	12:00:00 PM	Linda	Callahan	272 1189	Yes
			12:00:00 PM	Janice	Leverling	285 3412	Yes
			9:00:00 AM	Anmed	Singh	455 9482	Yes

Design Consideration for Database Systems

The design of the user interface of a database is as important as the design of the underlying database structure. The interface determines how people work with the database and impacts on the overall effectiveness of the system.

A good design should demonstrate the following:

Logical order. The components should be arranged in a way that is familiar for a user and should follow accepted patterns of grouping and sequence. For example, related fields should be grouped together; aligned headings, labels, text boxes and buttons should be used; and frequently used elements positioned in the most visible section of the screen.

Readability. The text choices for the interface should clearly communicate the operation of the interface to the user. For example, fonts should be kept consistent throughout the interface, all text should be at a readable size and white space should be used to ensure the user is not distracted.

Inclusivity. The design should cater for all users and for those people with disabilities. Design considerations should be made to ensure the full range of potential users are able to:

- Read or listen to the information on the page. For example, text alternatives are provided.
- Operate all the form's functions. For example, all functions can be carried out from the keyboard.
- Understand the content on the form. For example, the content is be predictable.

Navigation. The design of the user interface navigation system should provide a positive experience for the user. It should control the way the user makes decisions about locating the content and should be efficient and intuitive. Complex tasks should be broken down into logical multi-step processes. Important considerations include:

- Keep the navigation system consistent and within existing industry interface design parameters.
- Ensure that the user knows where they are and can backtrack easily if necessary
- Keep the navigation system simple and keep options to a minimum for effective navigation.
- Ensure menus are obvious and uncluttered.

Complete Activity Six

Database Documentation for the User

User documentation should be provided for both technical and non-technical purposes.

Technical documentation provides information to administer, develop and work with the database system. For example, this type of documentation includes database specifications, embedded comments, the data dictionary, backup and restore processes, database schema etc.

Non-technical user documentation describes how to operate the user interface and interact with each component in the database. It should be well organised, include notes and warnings and be written at a level that the user understands. For example, this type of documentation describes how to open and close the database, how to input data, how to find information in the database and how to output information in the desired format.

Data dictionary

A data dictionary is used to record the details about every item of data that is used in a database. This is an important part of the database user documentation. A data dictionary will record information about data such as details of name, the data type, field length or size and any contraints. It is common practice to maintain a data dictionary as a separate database.

An example of a data dictionary for an employee table is show below:

Element Name	Data type	Size/Format Default	Constraints	Description
Employee_num	Text	6 characters	Digits 0 – 9	Unique employee ID
Surname	Text	Up to 40 characters		
First Name	Text	Up to 40 characters		
Start_date	Date/Time	dd/mm/yyyy		Date employment commenced
Security_status	Text	1 characters	A, B, C or D	Default is D
Employee_address	Text	Up to 120 characters	3 lines	Address is a single field.
Photo	Object		Optional	Restricted to 5MB

Complete Activity Seven

Designing a Database

There are two methods that can be used to design a database: creating an **entity relationship diagram** or using a process called **normalisation**.

- The entity relationship diagram method involves modelling the data and its structure using diagrams. This approach is often used where existing methods of data handling do not exist or are not completely defined.
- The normalisation method is used where the attributes of data are known, for example paper-based information already exist in a manual data system. Normalisation is the steps or rules that can be used to derive sets of data suitable to be placed into tables in a relational database.

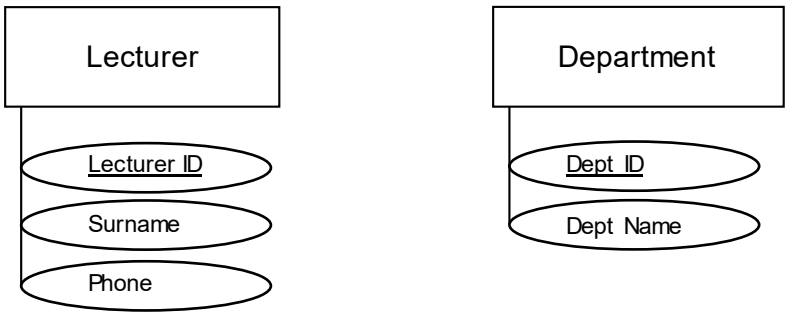
Creating an Entity Relationship Diagram

Below are the steps that can be used to create an entity relationship diagram to model a database, and then derive tables from that model:

Step One

The first step is to identify the entities that exist within a system. An entity is something distinguishable that you wish to record data about. Entities commonly are people, places, events, objects or concepts. Each entity has a set of attributes associated with it. One attribute (or a combination of attributes) is required to uniquely identify each instance of an entity. This is known as the key attribute and is underlined. Ultimately the entities you define will become the tables in your database. Each instance of an entity will become a record in the table. Each attribute will become a field.

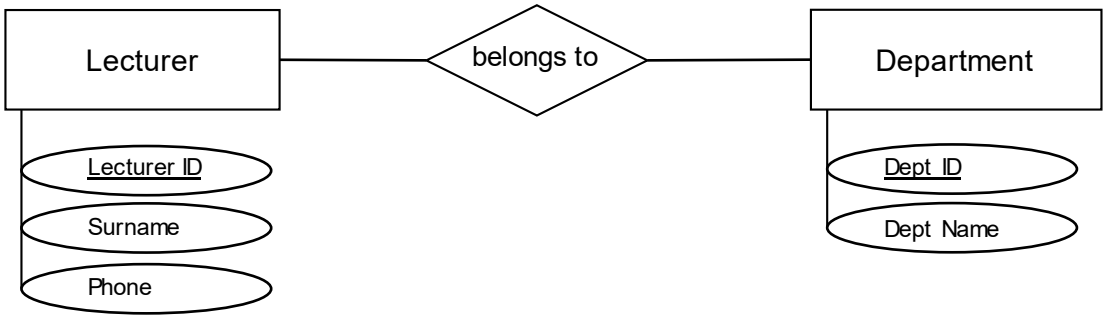
For example, Tuart College might identify two entities: Lecturer and Department. Here are how the two entities would be shown diagrammatically:



Step Two

The next step is to define the relationships that exist between entities. A relationship is what links entities together.

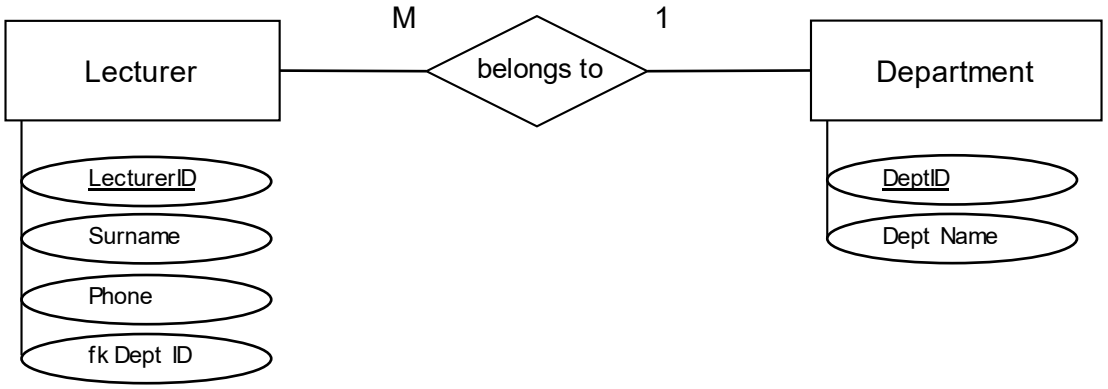
For example, a relationship exists between a lecturer and a department. A lecturer always belongs to a department and a department has lecturers working in it.



Step Three

This step involves defining the type of relationship, called **cardinality**, that exists between each entity. There are three types of relationships that can exist:

One to Many. In this relationship, each instance of one entity can be related to more than one instance of another entity. A one to many relationship is represented diagrammatically by a M at the many end of the relationship line. For example, a lecturer belongs to one department but a department can have many lecturers:

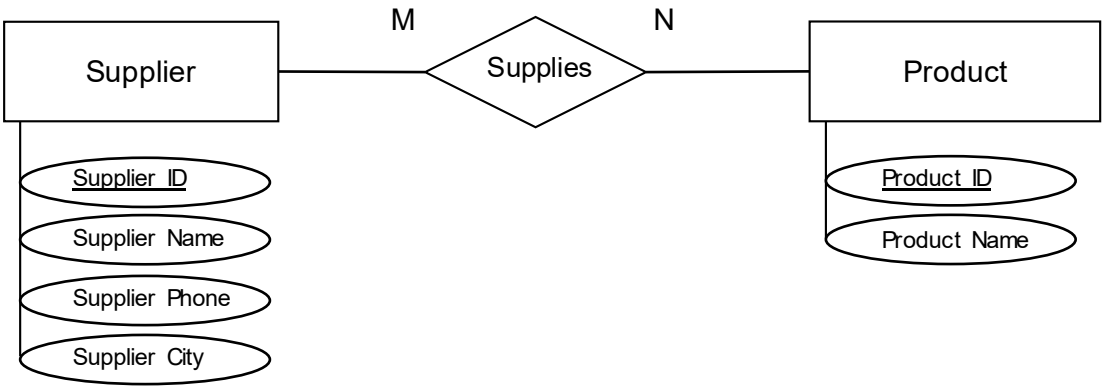


One to Many relationships are bi-directional, and can be traversed in either direction. For example, it is possible to find all the lecturers in a department or to find the department of a particular lecturer.

Once the cardinality has been represented, the foreign key attribute is added to one entity. In this example, the foreign key is added to the Lecturer entity as each lecturer can only ever work for one department.

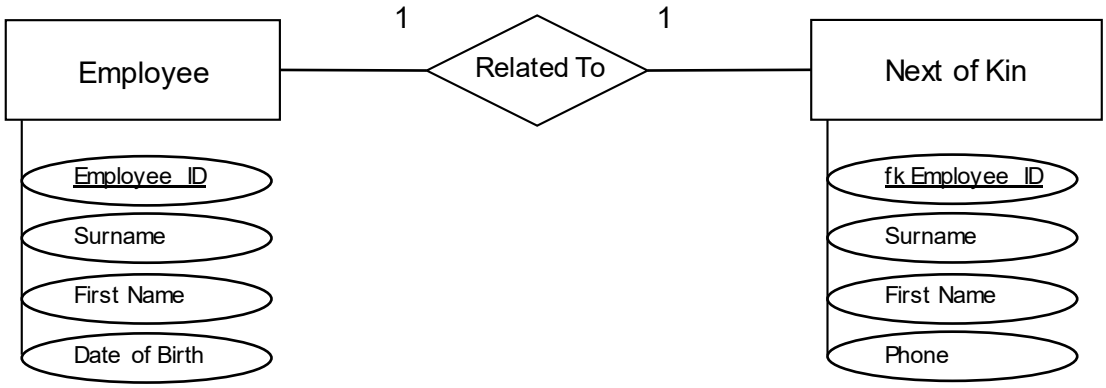
Many to Many. In this type of relationship, each instance of one entity can be related to more than one instance of another entity, and vice-versa. For example, suppose two entities are identified, suppliers and products. The relationship between these entities is many to many since:

- a supplier could supply more than one product
- a product could be supplied by more than one supplier.



As this relationship is not currently 1 to M, in its current form a foreign key is not able to be added to either entity.

One to One. In this relationship, for every record in one table there is exactly one record in another table. For example, to record details of the next of kin of employees, a one to one relationship would be required.

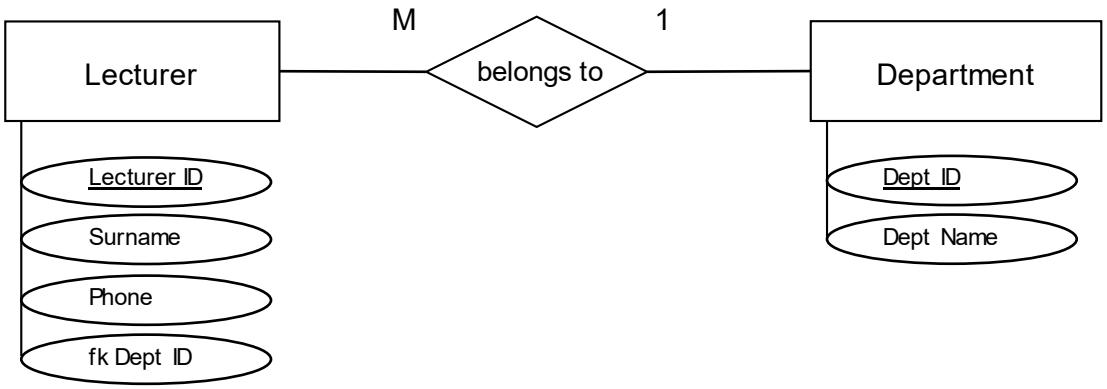


In a 1 to 1 relationship, the foreign key is the primary key of the second entity.

Stage Four

Entities and relationships can now be used to define the table structure of a database. Foreign keys are used to define the link between entities (tables).

For example, the one to many relationship between lecturers and departments would be converted as follows:



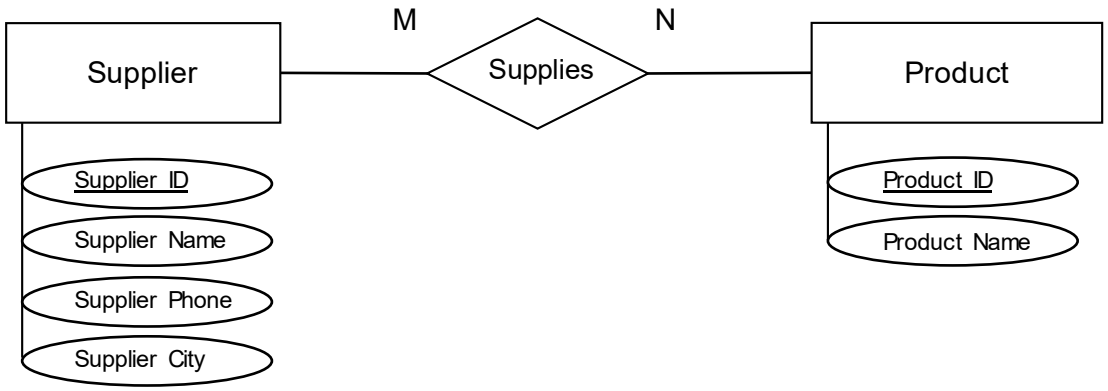
Lecturer ID	Surname	Phone	Fk Dept ID
1	James	9344 2222	1
2	Black	9222 4444	1
3	Smith	9111 6666	3

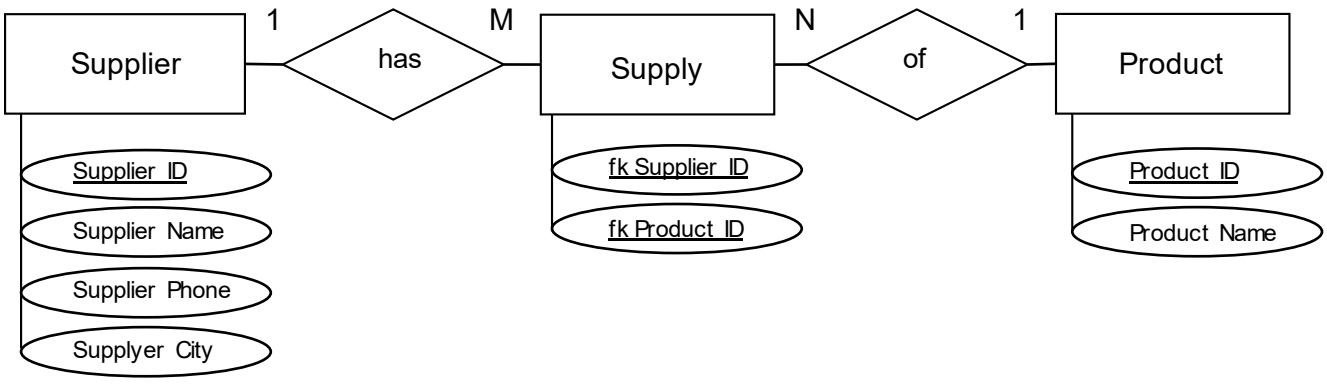
M

Dept ID	Dept Name
1	Computing
2	Maths
3	English

1

The many to many relationship between suppliers and products could be converted to two one to many relationships as follows:





The following table structure would be implemented.

Supplier ID	Supplier Name	Supplier Phone	Supplier City
1	Jones	93435656	Perth
2	Black	94353345	Perth
3	Smith	92342333	Albany

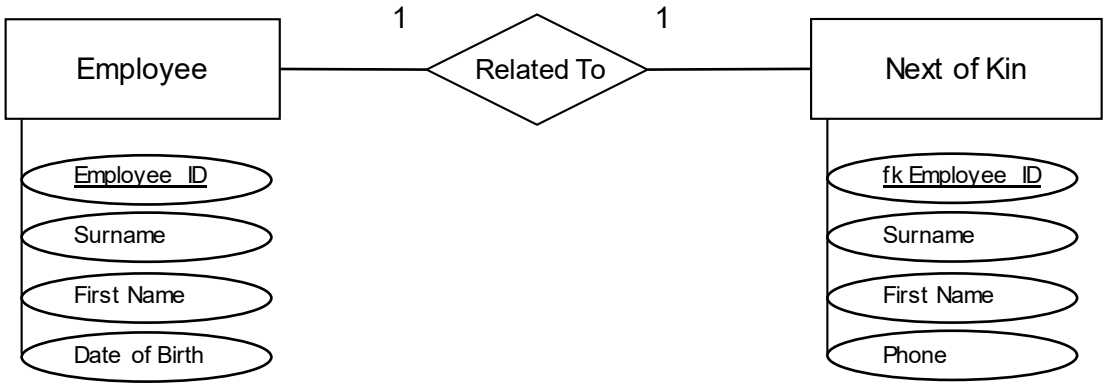
fSupplierID	fProductID
1	2
1	3
2	2

Product ID	Product Name
1	Oriental Tea
2	Strong Coffee
3	Ice Milk

1 M N 1

Implementing many to many relationships in a database always requires the creation of two, one to many relationships linked to an intermediate table containing two foreign keys.

The one to one relationship between employees and next of kin would be converted as follows:



Employee ID	Surname	First Name	Date of Birth
1	James	Mark	14/12/95
2	Black	Carol	27/03/90
3	Smith	Julie	04/08/97

fk Employee ID	Surname	First Name	Phone
1	Williams	Linley	9344 2222
3	Jackson	Isabella	9222 4444

1 1

In the above example we use two tables because not every employee may have a next of kin.

Complete Activity Eight

Normalisation

Normalisation is the process of decomposing a relation into a number of smaller relations suitable for implementation in a relational database.

Un-Normalised Data

Un-normalised data does not have single values stored in each cell in a table or has the same field name repeated in a table. For example:

Branch Name	Phone	Staff
South Perth Office	95449999	Nancy Jones, Janice Kent, Robert Nguyen, Linda Morris
Como Office	93337444	Andrew Fuller, Jonathan King
Bunbury Office	96456782	Ghina Sarraf

Branch Name	Phone	Staff 1	Staff 2	Staff 3	Staff 4
South Perth Office	95449999	Nancy Jones	Janice Kent	Robert Nguyen	Linda Morris
Como Office	93337444	Andrew Fuller	Jonathan King		
Bunbury Office	96456782	Ghina Sarraf			

First Normal Form (1NF)

Data is in First Normal Form (1NF) when each attribute (field) has only atomic values. Any data that is placed into a table with correctly identified fields (no repeats), single values in each cell and unique records will generally be in first normal form. For example:

Staff ID	Staff Surname	Staff First Name	Branch Name	Branch Phone
1	Jones	Nancy	South Perth Office	95449999
2	Fuller	Andrew	Como Office	93337444
3	Kent	Janice	South Perth Office	95449999
4	Sarraf	Ghina	Bunbury Office	96456782
5	Nguyen	Robert	South Perth Office	95449999
6	King	Jonathan	Como Office	93337444
8	Morris	Linda	South Perth Office	95449999

Problems

- Data repetition/redundancy in records.
- Update, deletion, insertion anomalies.
- Poor storage utilisation.

Second Normal Form (2NF)

A relation is in second normal form (2NF) when each **non-key field** is **functionally dependent** on the **primary key**.

Functional Dependency: For each key there will be precisely one matching value in the non-key field.

Problems

- Data repetition/redundancy in records.
- Update, deletion, insertion anomalies.
- Poor storage utilisation.

Third Normal Form (3NF)

A set of relations is in third normal form (3NF) when each non-key field is **fully functionally dependent** on the primary key.

Full Functional Dependency: Each non-key field will be functionally dependent only on the key and not on any other field.

For example, the tables below are in 3NF.

Staff

Staff ID	Staff Surname	Staff First Name	fk Branch ID
1	Jones	Nancy	1
2	Fuller	Andrew	2
3	Kent	Janice	1
4	Sarraf	Ghina	3
5	Nguyen	Robert	1
6	King	Jonathan	2
7	Morris	Linda	1

Branch

Branch ID	Branch Name	Branch Phone
1	South Perth Office	95449999
2	Como Office	93337444
3	Bunbury Office	96456782

The result is a set of relations that can now be implemented in a relational database program such as Microsoft Access.

Complete Activity Nine

Types of Databases

Distributed databases and centralised databases are two systems for organising a database across different locations.

In a **distributed database**, data is stored at different locations that are connected together by a network. Each location is able to store and process its own data. Locations work together via the network as if data is stored on a single site. Distributed databases are managed by software called a Distributed Database Management System (DBMS)

A **centralised database** has the DBMS located at one central location. All data storage and processing takes place at this one location. Users at different locations either on site or off site can connect to the central database via the network using workstations, but do not store data locally.

Distributed Databases Advantages	**Centralised Databases Advantages**
There is local control over the database (security, data integrity, maintenance) There is no reliance on a central site. For example, bottlenecks are reduced. Faster processing is possible, for example queries.	All data is processed at one location making it easier to manage data integrity and data security. The system is simpler to administer as there is one DBMS at one location.
Distributed Databases Disadvantages	**Centralised Databases Disadvantages**
The system requires more complex software and higher costs to administer. There are increased data processing costs such as consistency management and query processing.	The system is susceptible to bottlenecks in accessing data, particularly query processing. There are increased system vulnerabilities, for example if the central database fails, all sites are affected.

Data Warehouses

It is important for managers to have access to a wide range of relevant information to inform their decision making. An organisation's database contains a wealth of data about the organisation but it is not always easy to extract useful information from this data. A **data warehouse** is a large database system that integrates with the operational database and separately stores historical and relevant information that can be used for analysis and reporting.

The process of searching for patterns in the large set of data held in the data warehouse is called **data mining**. Data mining looks for patterns in the data and produces relevant insights that can be used to provide useful information for the organisation.

Departments within an organisation may have special needs for information and may want to have the ability to directly manage their own information needs. A **data mart** is a sub-set of the data warehouse that is designed to align with the specific business needs of part of the organisation, for example sales or marketing. Often the data mart is a partitioned segment of the organisation's data warehouse.

There are **ethical implications** that arise from the use of data warehouses, data mining and data marts. These include:

- Control of privacy: Data mining makes it possible to analyse routine business transactions across large data sets. This can provide insights that may compromise the privacy of individuals. For example, personal buying habits or personal preferences.
- Misinterpretation of data: The data may contain errors, may be incomplete or out of date. As a result, it is possible to make poor decisions that can have a detrimental impact on people's lives.

Data Security

The data in a database needs to be protected from unauthorised access, from data loss, data corruption and incorrect use of data.

The methods to protect data and ensure **data security** include:

- **Identification and access control.** Users are identified by a username and password to prove their identity and to control their access. Usernames are allocated to accounts that belong to one or more user groups. A user's access to parts of the database is determined by the access permissions of the user group they belong to.
- **Encryption of data.** This involves encoding data so that only authorised access is possible using a key to decode the data. Encryption protects the data from being read indirectly by applications other than the DBMS. It also protects data during communication processes.
- **Disaster recovery.** This is the methods for restoring information that has become damaged or lost due to incorrect use or system failure. This is often referred to as backup procedures.

The Role of Open Systems

Open systems play an important role in the development and management of databases. An open system allows different vendors and developers to develop products that can communicate and share access to each other's systems.

It is important that data from different DBMS proprietary applications is able to be shared by other applications. **Open Database Connectivity (ODBC)** is a standard Application Programming Interface (API) specification designed to make it possible to access the data managed by any DBMS. This is achieved by inserting a middle layer that translates communication between the DBMS and the querying software application into a common language.

Online databases are databases that are implemented using the internet; the database is stored on a server and accessed via a web interface. The advantage of an online database is that a standard browser can be used as the user interface and is universally available.

A **data driven website** is a website that dynamically assembles the requested web pages using live data held in a database. This is different to a static website where the requested pages always contain the same prebuilt content. Open system connectivity is important to enable the communication between the data driven website and the underlying database, even where both products are developed by different vendors.

Complete Activity Ten

Department of Education Western Australia 2020

Computer Science ATAR Year 12

Resource Package

Managing Data

Activities

Instructions to Students

This resource package provides students with learning materials for the Computer Science ATAR Year 12 course. The package focuses on content from the Managing Data content area in the 2020 Year 12 Syllabus. The content covered from the syllabus is listed on Page 2 of the Notes document.

This package is designed to support the program students are completing at their school. If feedback is required when completing this package, students should consult their teacher.

This resource package consists of two parts:

- The **Notes** document provides an explanation of syllabus content concepts. This section is designed to develop the knowledge component of the syllabus.

  The Notes document is in PDF form.

- The **Activities** document provides an opportunity for students to actively engage with Computer Science content. The activities are designed to develop both the knowledge and the skills components of the syllabus.

  The Activities document is provided in both PDF and Microsoft Word format. Students can print the PDF version or work directly on the editable Microsoft Word document.

Students should read one section at a time of the **Notes** document. The associated **Activities,** indicated in blue text at the end of the section, should be completed before moving onto the next Notes section. If students are completing the Activities in Word, it is recommended that they open both the Notes and the Activities documents and switch between reading the content and completing the associated Activities.

It is recommended that students further investigate concepts covered in this resource package by conducting their own research using the internet. There are many resources, for example on YouTube, that will further develop understandings of the Computer Science concepts.

Activity One

1. Use the internet to find two definitions for the term database. Ensure you find different definitions to the one provided in your notes.

Definition One	
Definition Two	

2. Use the internet to find three interesting examples of commonly used **online databases**.

Example One	
Example Two	
Example Three	

3. Explain in your own words how data is different from information.

4. Add an extra **field** called Pet Description to the table below. Make up some pet information and add two pet **records**.

Pet Name	**Pet Age**	**Pet Breed**	

5. It is possible to use Microsoft Excel as simple single table database. Explain why Excel **is not** suitable for more complex database solutions.

Activity Two

1. Use the internet to read more about Database Management Systems (DBMS). From your readings, list and briefly describe four common functions provided by a DBMS.

	Name	Brief Description
Function 1		
Function 2		
Function 3		
Function 4		

2. Use the link https://db-engines.com/en/ranking to look at the ranking of popularity of Database Management Systems. Indicate the current ranking of each of the following:

DBMS	Ranking
Microsoft Access	
IBM DB2	
Oracle RDBMS	
FileMaker	

3. View the first **eight minutes** of the video https://www.youtube.com/watch?v=PBhftKTmdHI on YouTube. The video describes the process of creating a database and table in Microsoft Access. List the steps needed to create the **table** in Microsoft Access. The first steps to create the database have been listed for you. Add further rows as required.

Open Access
Choose Blank Desktop Database
Name and create the database
Close the default table

4. A database consultant is creating a database for a car hire firm. She has identified an entity called Vehicle. This entity will become a table in the database.

   Complete the Vehicle table below by creating three appropriate fields for a car (make up the fields that you think a car would have). Indicate (underline) which field will become the primary key. Complete two records (make up the car details).

Car Registration			
1APO733			
1ZSP733			

5. Explain why a composite primary key is necessary in some tables?

6. In the relation definitions below, highlight each of the following by changing the font colour:
   - the entity (Green)
   - the primary key (Blue)

   Student: Student ID, Student First Name, Student Surname, Student Address

   Drivers Licence: Licence Number, Surname, First Name, Address, Date of Birth, Expiry Date

   House: Street number, Street Name, Suburb, House type, Number of Bedrooms

7. Watch the YouTube video https://www.youtube.com/watch?v=7lJS5tklOrE. Explain how you would create a primary key in Microsoft Access.

   What key would you press if by a mistake you enter a duplicate primary key value into a field in a Microsoft Access table?

Activity Three

1. Draw a line (Insert|Shapes on the Microsoft Word toolbar) to show the link between the foreign key and the primary key in the two tables below. Also add a new student 3443 Paul Bailey to the student table. Paul is in Wollaston.

Student ID	Student First Name	Student Surname	fk House Group
2343	John	Ng	Bilu
2433	Adrian	Wilson	Wollaston

House Group Name	House Group Coordinator	House Group Colour
Bilu	Mrs Jones	Red
Wollaston	Mr Taylor	Blue

2. Watch video https://www.youtube.com/watch?v=NtRAyS0LLlk on YouTube.

   What are the names of the two tables that are linked in the video?

   What is the name of the foreign key field in the relationship?

   What is the name of the primary key field in the relationship?

Under the Database Tools tab on the Microsoft Access toolbar, what is the name of the button you click to create a relationship between two tables? If you know how to, copy and paste an image of the button into your answer below.

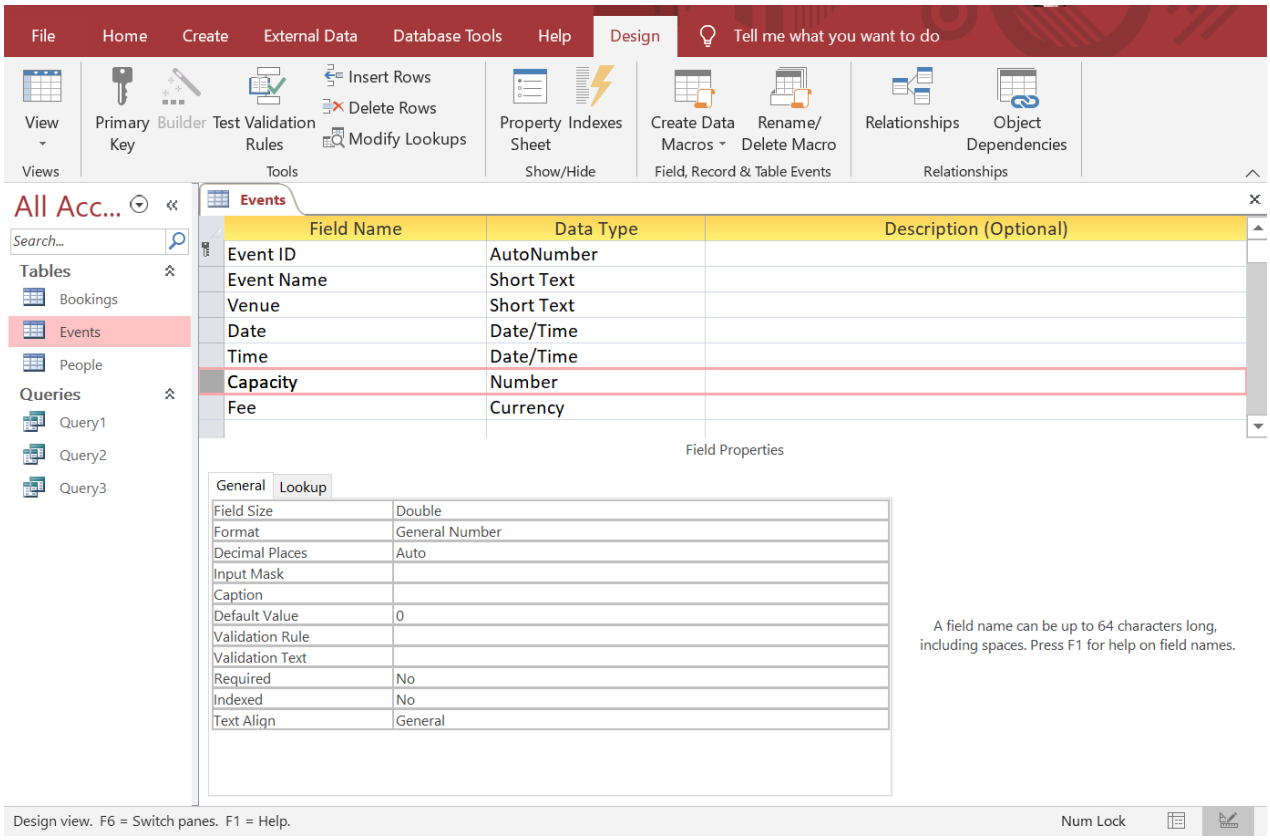
What is the purpose of the Show Table box?

Describe how you create the link between the foreign key and the primary key.

How do you enforce referential integrity in the relationship?

Does Microsoft Access provide a way of creating cascade update and cascade delete? Explain your answer.

3. The image below shows Design View for a Microsoft Access table called Events. The field Capacity is currently selected.



Used with permission from Microsoft.

What is the primary key in the table Events?

What data types are used in the table Events?

The property Field Size indicates that a long integer has been used to store the capacity of the event. What is a long integer?

What is the default value for the field Capacity? What does this mean?

What is the validation rule used for the field Capacity? Explain this validation rule.

Activity Four

1. In the example below, a coffee wholesale company called The Right Brew stores information about the coffee products they sell and the coffee manufacturers who produce the coffee products. Currently the information is stored in a **single** table. You are a consultant and have identified three problems that occur when working with this data: update anomaly, insert anomaly and delete anomaly.

   Update anomaly: Crabtree & Smith, Ltd have changed their box number to Box 168. Update the table below. How many records have been updated? Why are details about Crabtree & Smith stored more than once?

   Insertion anomaly: Another product called Fraser Beans (unblended coffee), supplied by The Coffee Bean, is now being sold. Add the data to the table below. What information did you have to repeat that already exists in the table?

   You have found a new coffee manufacturer called Mark & Co, located at 21 Broom Street, Melbourne VIC. You don't have any coffee product information yet, but you need the manufacturer in the database. Add the manufacturer to the table. How many blank cells result in the table after you added Mark & Co?

   Deletion anomaly: African Raw Beans is no longer sold. Place a line through this entire record (use ~~strikethrough~~). If this record is removed, what other information will be lost?

Product Name	Blended Product	Manufacturer	Address	State	City
Eliza House Blend	Yes	The Coffee Bean	250 Edwards Street	WA	Perth
Roseberry Raw Beans	No	The Coffee Bean	250 Edwards Street	WA	Perth
American Roast	Yes	Fortnum & Mason	1481 3rd Street	CA	San Francisco
Mexican Roast	No	Crabtree & Smith	Box 167	NSW	Sydney
Alex Blend	Yes	The Coffee Bean	250 Edwards Street	WA	Perth
Bess House Blend	Yes	The Coffee Bean	250 Edwards Street	WA	Perth
French Raw Beans	No	Crabtree & Smith	Box 167	NSW	Sydney
Bill's House Blend	Yes	Crabtree & Smith	Box 167	NSW	Sydney
African Raw Beans	No	Tea & Coffee Co	200 St Kilda Rd	VIC	Melbourne

2. You have designed a relational solution for the above information that uses the two tables. Fill in the incomplete the *fk Manufacturer ID* field (three values are missing) in the Product table and then answer the questions below.

Product

Product ID	Product	Blended Product	Fk Manufacturer ID
1	Eliza House Blend	Yes	1
2	Roseberry Raw Beans	No	
3	American Roast	Yes	2
4	Mexican Roast	No	3
5	Alex Blend	Yes	
6	Bess House Blend	Yes	1
7	French Raw Beans	No	
8	Bill's House Blend	Yes	3
9	African Raw Beans	No	4

Manufacturer

Manufacturer ID	Manufacturer	Address	State	City
1	The Coffee Bean	250 Edwards Street	WA	South Perth
2	Fortnum & Mason, Ltd.	1481 3rd Street	CA	San Francisco
3	Crabtree & Smith, Ltd.	Box 167	NSW	Sydney
4	Tea & Coffee Co	200 St Kilda Rd	VIC	Melbourne

Crabtree & Smith, Ltd have changed their box number to Box 168. Update the Manufacturer table. How many records did you need to update? How is this an improvement over the single table database?

Another product called Fraser Beans (unblended coffee), supplied by The Coffee Bean, is now being sold. Add the information to the Product table (the Product ID will be 10). How is this an improvement over what needed to be added to the single table database?

You have found a new coffee manufacturer called Mark & Co, located at 21 Broom Street, Melbourne VIC. You don't have any coffee product information yet, but you need to have the manufacturer in the database. Add this manufacturer to the table. The Manufacturer ID will be 5. How is this an improvement over adding the manufacturer to the single table database?

African Raw Beans is no longer sold. Place a line through this entire record (use ~~strikethrough~~). How is this an improvement over the removing the record in the single table database?

Activity Five

1. What are the three advantage of using queries for finding information compared to searching for information directly in tables using filters?

1.
2.
3.

2. Watch the YouTube video https://www.youtube.com/watch?v=LUL1nnxUz_c. The query in the video uses fields from a table called Customers. The records in the query are sorted. Two criteria are used to show only customers who are from the city Raleigh or have a zip code 27513.

   List the steps needed to create the Microsoft Access query discussed in the video. The first two steps have been created for you. Add further rows as required.

Choose the Create tab from the toolbar
Click on the Query Design button

3. Use the two Microsoft Access tables below, Branch and Staff, to answer the following questions about five queries that have been created from this data.

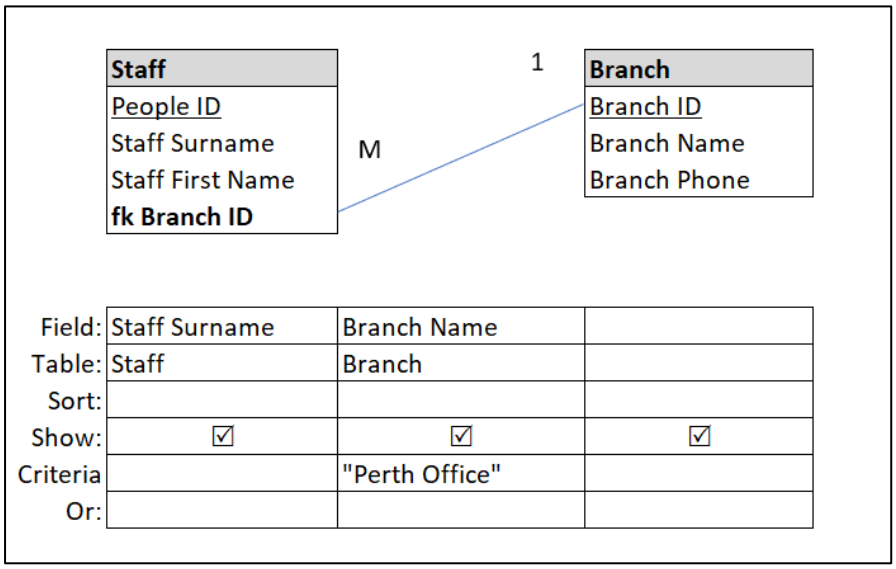
Branch

Branch ID	Branch Name	Branch Phone
1	South Perth Office	95449999
2	Como Office	93337444
3	Bunbury Office	96456782

Staff

Staff ID	Staff Surname	Staff First Name	fk Branch ID
1	Jones	Nancy	1
2	Fuller	Andrew	2
3	Kent	Janice	1
4	Sarraf	Ghina	3
5	Nguyen	Robert	1
6	King	Jonathan	2
7	Morris	Linda	1

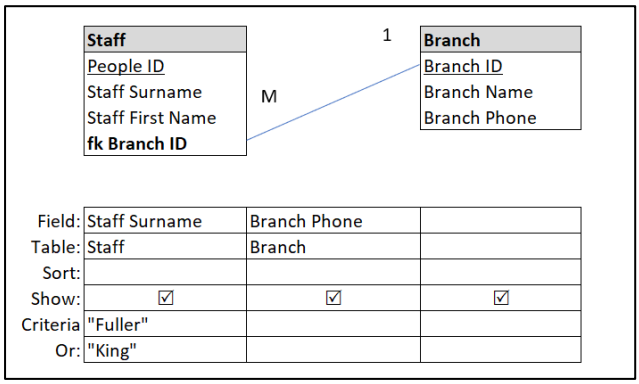
Query 1



Complete the last three records in the following table to show the output from Query 1:

Staff Surname	**Branch Name**
Davolio	Perth Office
Leverling	Perth Office

Query 2

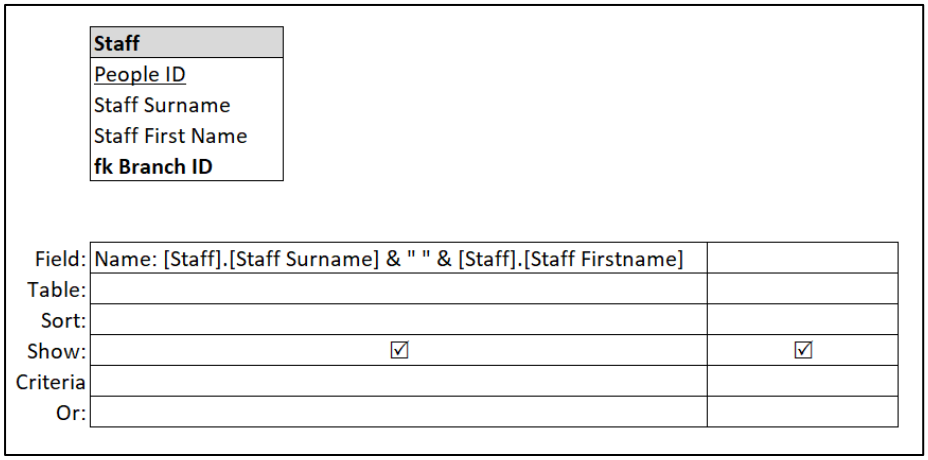


Field:	Staff Surname	Branch Phone	
Table:	Staff	Branch	
Sort:			
Show:	☑	☑	☑
Criteria	"Fuller"		
Or:	"King"		

Complete the following table to show the output from the Query 2:

Staff Surname	Branch Phone
	95449999
	93337444

Query 3



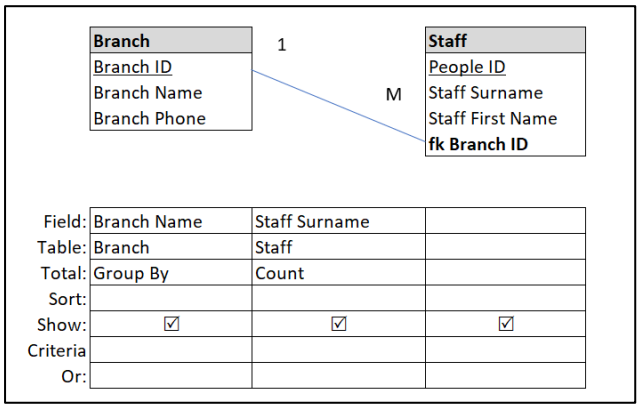
Field:	Name: [Staff].[Staff Surname] & " " & [Staff].[Staff Firstname]	
Table:		
Sort:		
Show:	☑	☑
Criteria		
Or:		

Output from Query 3:

Name
Davolio Nancy
Fuller Andrew
Leverling Janice
Peacock Eliza
Buchanan Robert
Suyama Michael
King Jonathan
Callahan Linda
Dodsworth Annabella

What type of advanced query is Query 3? Explain what this query does.

Query 4



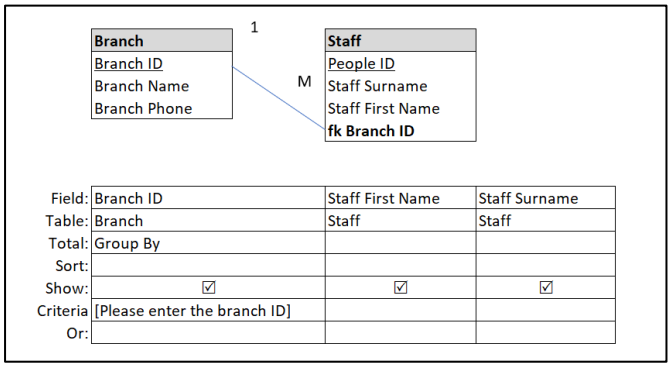
Field:	Branch Name	Staff Surname	
Table:	Branch	Staff	
Total:	Group By	Count	
Sort:			
Show:	☑	☑	☑
Criteria			
Or:			

Output from Query 4:

Branch Name	CountOfStaff Surname
Bunbury Office	1
Como Office	3
Perth Office	5

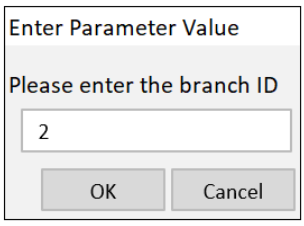
What type of advanced query is Query 4? Explain what this query does.

Query 5



Field:	Branch ID	Staff First Name	Staff Surname
Table:	Branch	Staff	Staff
Total:	Group By		
Sort:			
Show:	☑	☑	☑
Criteria	[Please enter the branch ID]		
Or:			

Output from Query 5:



Branch ID	Staff Surname
2	Fuller
2	Peacock
2	Suyama

What type of advanced query is Query 5? Explain what this query does.

4. The update, delete and append queries are called Action queries. What is meant by **action** and how are action queries different form ordinary select queries (such as the ones above).

5. A database consisting of two tables has been created to store information about hotels and rooms.

Hotels

HotelNum	HotelName	HotelStreet	HotelCity
H1	Grosvenor	21 Richmond Hill	London
H2	Esplanade	9 Seaview Terrace	Brighton
H3	Ritz	Trafalga Square	London
H4	Carlton	32 Bond Street	London
H5	Parmelia	St Georges Terrace	Perth
H6	Crystal Palace	31 London Street	Perth
H7	Hyatt	49 Hyde Road	London
H8	Pheonix	1147 High Road	London

Rooms

RoomNum	fkHotelNum	RoomType	RoomPrice
101	H1	Single	$60.00
102	H1	Single	$65.00
103	H1	Double	$80.00
104	H1	Double	$80.00
211	H2	Double	$50.00
212	H2	Family	$55.00
32	H3	Family	$40.00
33	H3	Family	$38.00
401	H2	Single	$30.00
402	H2	Double	$35.00
403	H2	Family	$38.00

Using the structure and contents of the two tables write the SQL statements for the following queries:

List the HotelName and HotelNum for all hotels.

```
SELECT

FROM
```

List the HotelName and HotelStreet for hotels in London.

```
SELECT

FROM

WHERE
```

List the HotelName and HotelCity for hotels in London or Brighton.

List the RoomNum, fkHotelNum, RoomType and RoomPrice for the rooms that are less than $60 per night.

List the RoomNum, fkHotelNum, RoomType and Room Price for family rooms that cost $40 per night.

List the RoomNum, HotelNum, HotelName and RoomType for all hotels.

Update the records in the table Rooms. Change rooms that currently have a price of $38 to a new price of $42 (Use the internet to find the syntax for an SQL update query).

Delete all the records for single rooms in the table Rooms (Use the internet to find the syntax for an SQL delete query).

Activity Six

1. What is the advantage of creating forms to access the data in tables in a database rather than working directly from the table?

Visit the Qantas flight booking page - https://www.qantas.com/au/en/book-a-trip/flights.html

Is this web page an example of a form? Do you think that this page is accessing a relational database? If so, provide an example of the information that the database would contain?

2. Watch the YouTube video on creating a report.

https://www.youtube.com/watch?v=1XUeGq80R5Q

What advantage is given in the video for using a report rather than a query to display information?

Can a report be used to edit data in the underlying table or query?

What two operations does the video demonstrate being applied to the report?

3. Visit W3.org using the link https://www.w3.org/WAI/standards-guidelines/wcag/glance/.

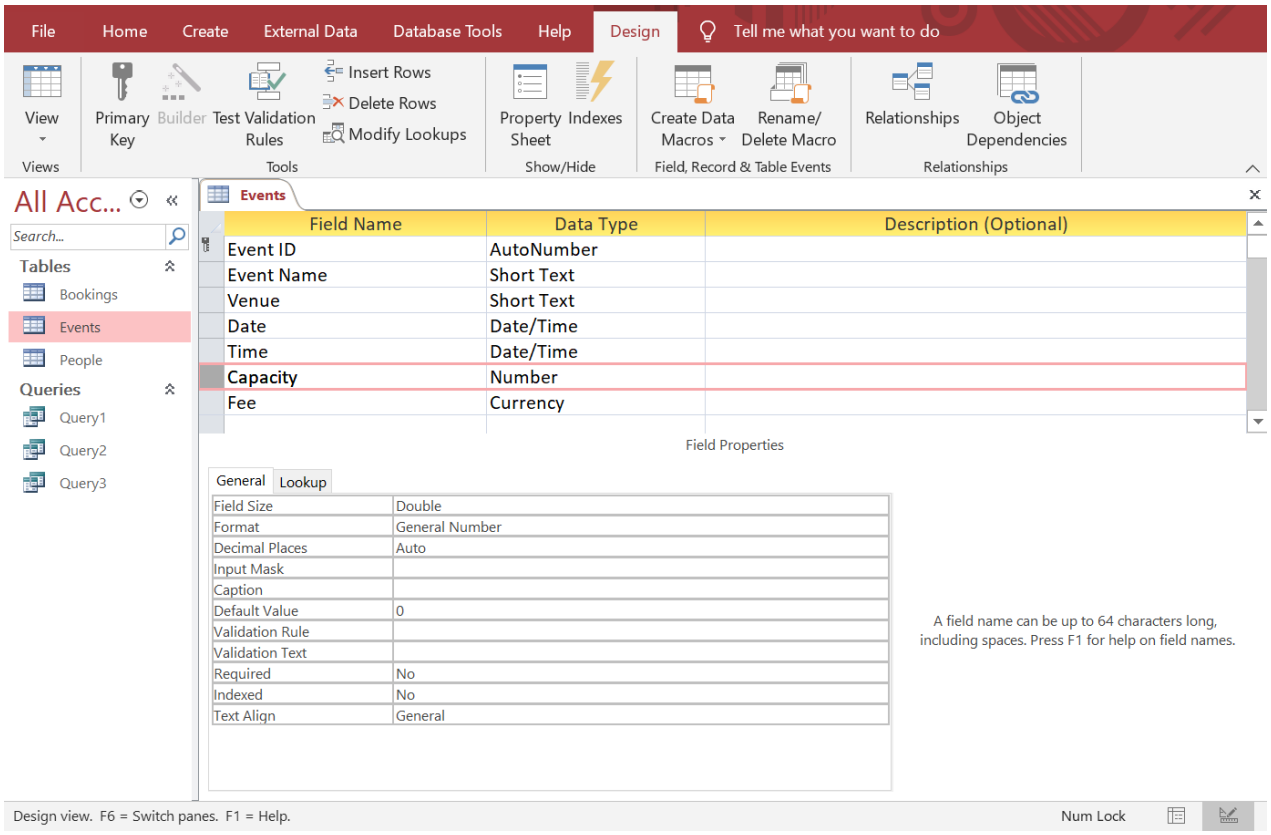
   What is the goal of the **World Wide Web Consortium (W3C) Web Accessibility Initiative (WAI)**? Hint: click on the About W3C WAI link at the top of the web page.

Under each of the four heading for web accessibility, briefly describe the ways that are recommended by the W3C for making content more accessible.

Perceivable	
Operable	
Understandable	
Robust	

Activity Seven

1. Complete the data dictionary for the Microsoft Access Events table shown below. Where not shown, you can make up values that you think would be appropriate for the fields.



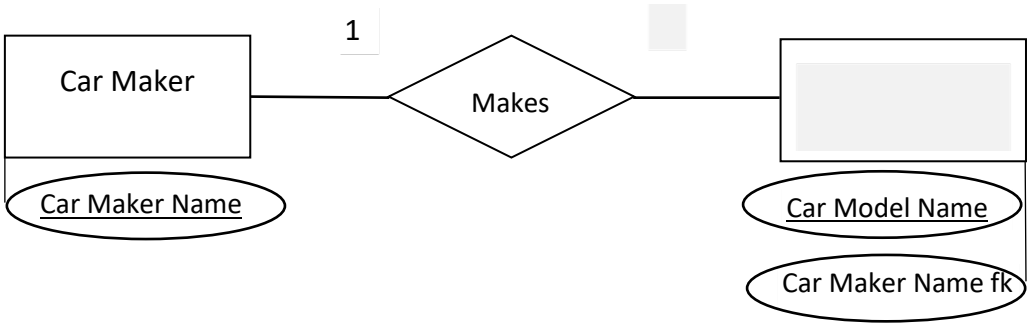
Used with permission from Microsoft.

Element Name	Data type	Size/Format Default	Constraints	Description
Event ID				

Activity Eight

1. Complete the ER diagram below to show the entity name and cardinality for car makers and car models. Car makers can make many models but a model is made by only car maker.

Car Maker Name	Car Model Name
Ford	Laser
Ford	Falcon
Toyota	Camry
Mitsubishi	Magna
Honda	Prelude
Holden	Commodore
Honda	Civic

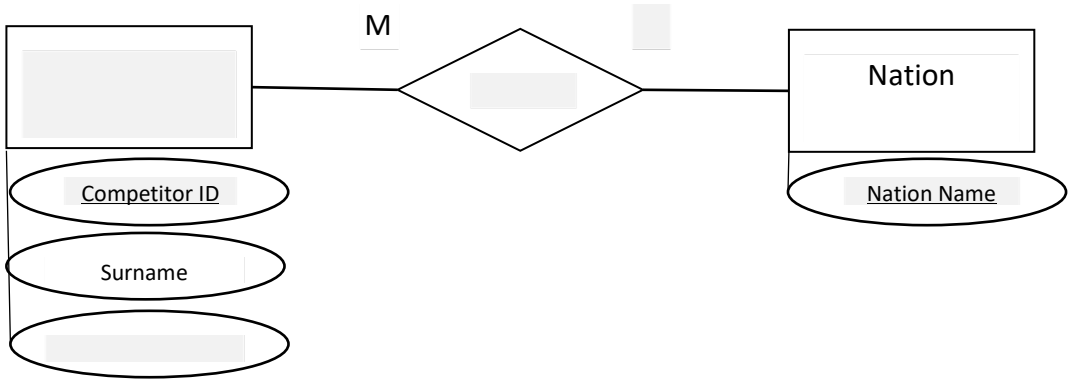


2. Complete the statements describing the relationship between software and computers.

A software title (e.g. Microsoft Access) can be installed on _____ computers.

A computer can have _____ software titles installed on it.

3. Use the table below to complete the ER diagram for competitors and the nation they represent. Show the entity name, relationship, primary keys, foreign keys, cardinality and attributes.



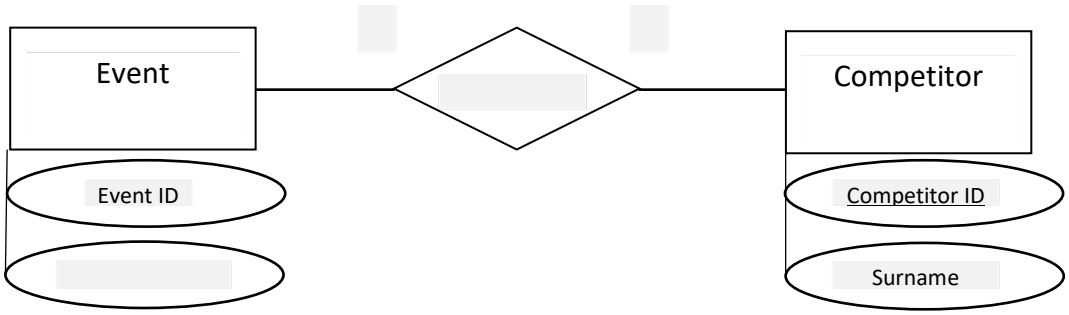
Competitor ID	Competitor Surname	Nation Name
1	Poll	Costa Rica
2	Beard	USA
3	Kasamatsu	Japan
4	Popov	Russia
5	Freeman	Australia
6	Klim	Australia
7	Saito	Japan

4. A swimming event can have many competitors. A competitor can compete in many events (eg 100m freestyle, 200m freestyle, 1500m freestyle)

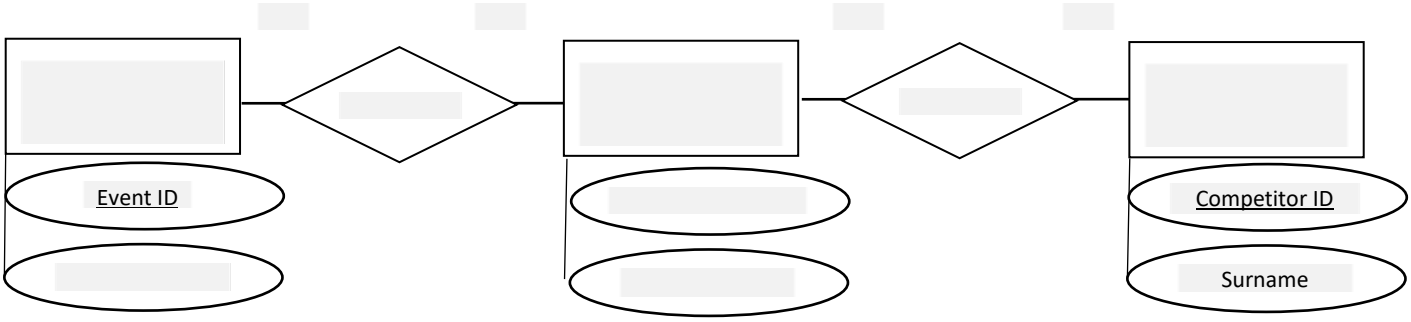
ID	Competitor	Event Name	Event Name	Event Name
6	Klim	Men's 100m Freestyle		Men's 1500m Freestyle
8	Thorpe		Men's 200m Freestyle	Men's 1500m Freestyle
4	Popov	Men's 100m Freestyle	Men's 200m Freestyle	
9	Perkins	Men's 100m Freestyle	Men's 200m Freestyle	
10	Lezac		Men's 200m Freestyle	Men's 1500m Freestyle
11	Schroeder	Men's 100m Freestyle		Men's 1500m Freestyle

Use the above un-normalised table of competitors and events to complete the ER diagram below.

- Complete the list of attributes below each entity.
- Underline the primary keys.
- Add the relationship name and the cardinality.



- Complete the diagram so that the many to many relationship is resolved. Indicate the foreign keys.



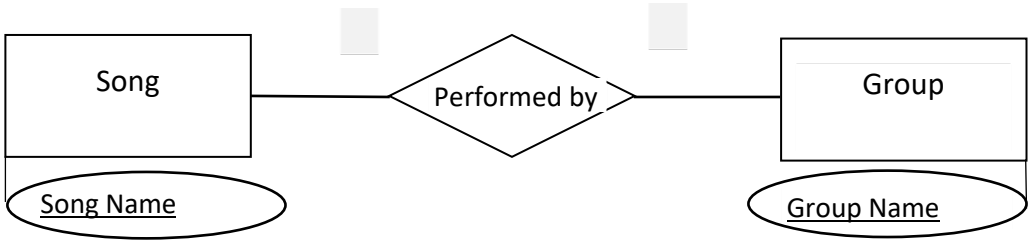
Complete the tables below to show the new relationships.

Event ID	Event Name
1	Men's 100m Freestyle
2	
3	

fEvent ID	fCompetitor ID
1	6
1	

Competitor ID	Surname
6	Klim
2	

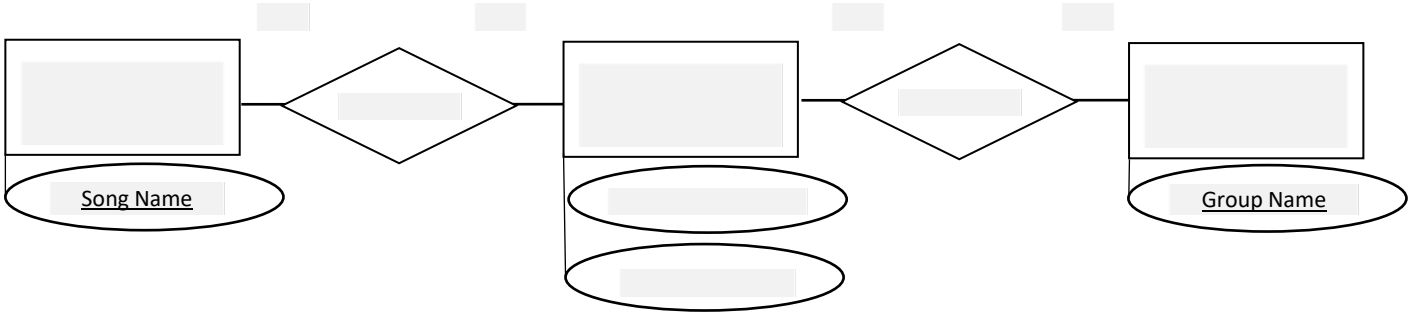
5. Complete the following E-R Diagram and the statement describing the relationships.



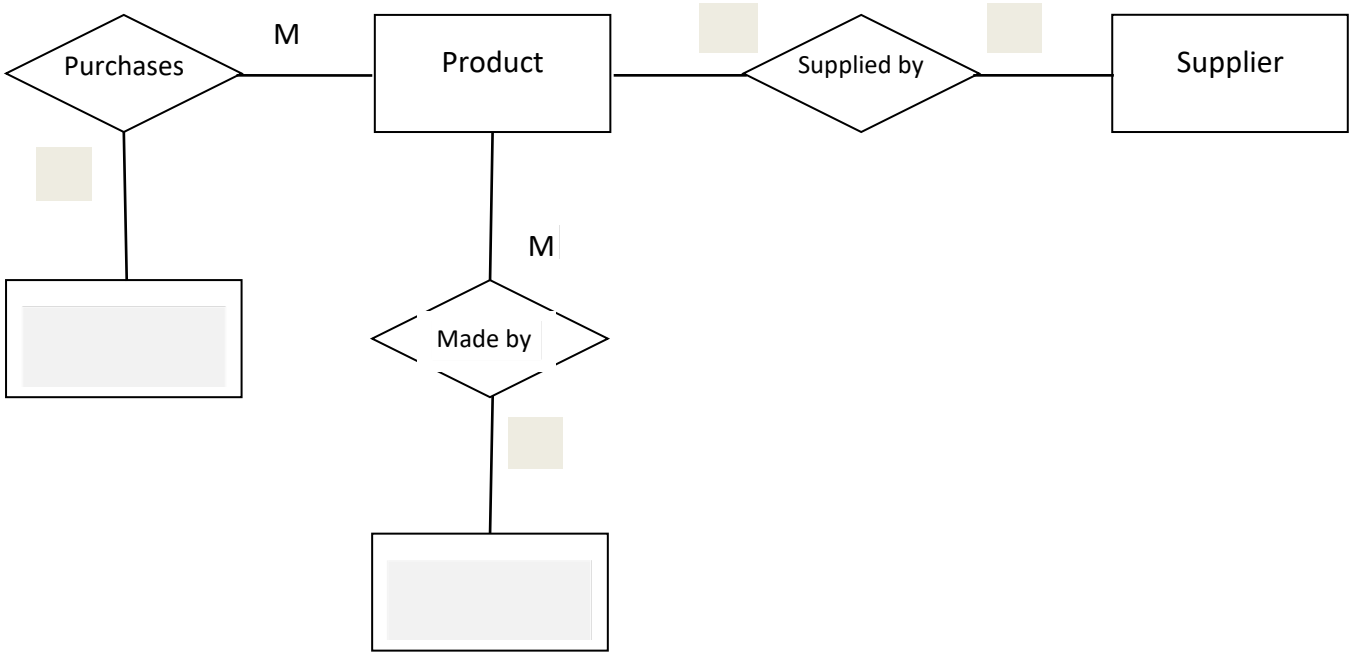
A song is performed by many groups.

A group can perform _____.

Complete the ER diagram below for implementation in a database.



6. A product (eg microwave oven) can be purchased by many customers. A product is manufactured by the one manufacturer. A product can be supplied by many suppliers. A supplier can supply many products. Complete the cardinality for following E-R Diagram based on the above description.



Redraw the diagram to resolve any problems with the many to many relationships. You may need to print this page and complete your diagram on paper.

List the cardinality, the attributes for each entity, underline the primary keys and indicate the foreign keys.

Customer

Customer ID
First Name
Surname
Phone

Supplier ID
Name
City
Phone

Supplier

Product

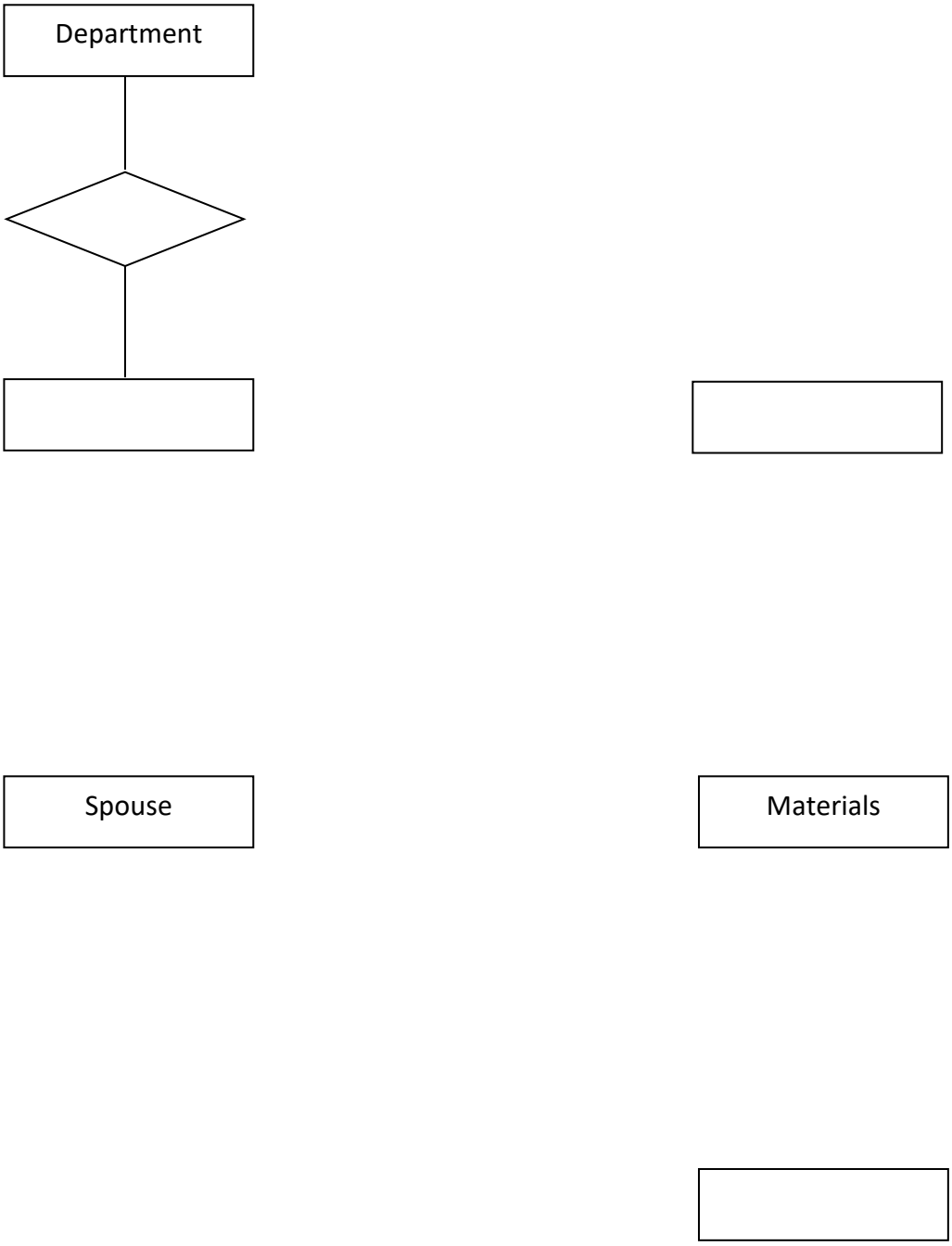
Product ID
Name

Manufacturer ID
Name
City
Phone

Manufacturer

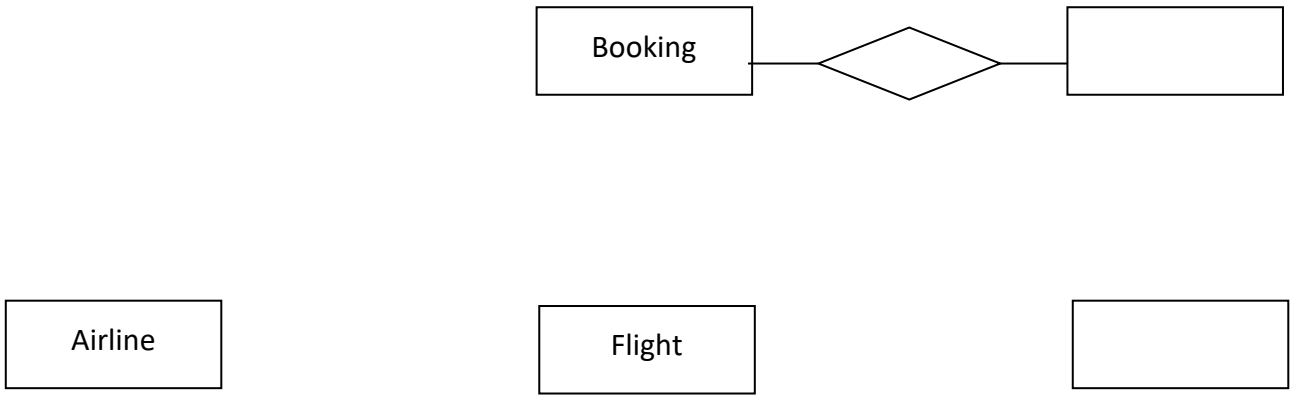
7. A Department has many Employees, an Employee works for only one Department. An Employee has only one spouse (husband or wife). An Employee can work on many Projects and each Project has many Employees. Many Materials are used on each Project and the Materials are supplied by many Suppliers. Each Supplier produced many types of Materials.

   Complete the following E-R Diagram based on the above description. Make up your own attributes. You may need to print this page and complete your diagram on paper.

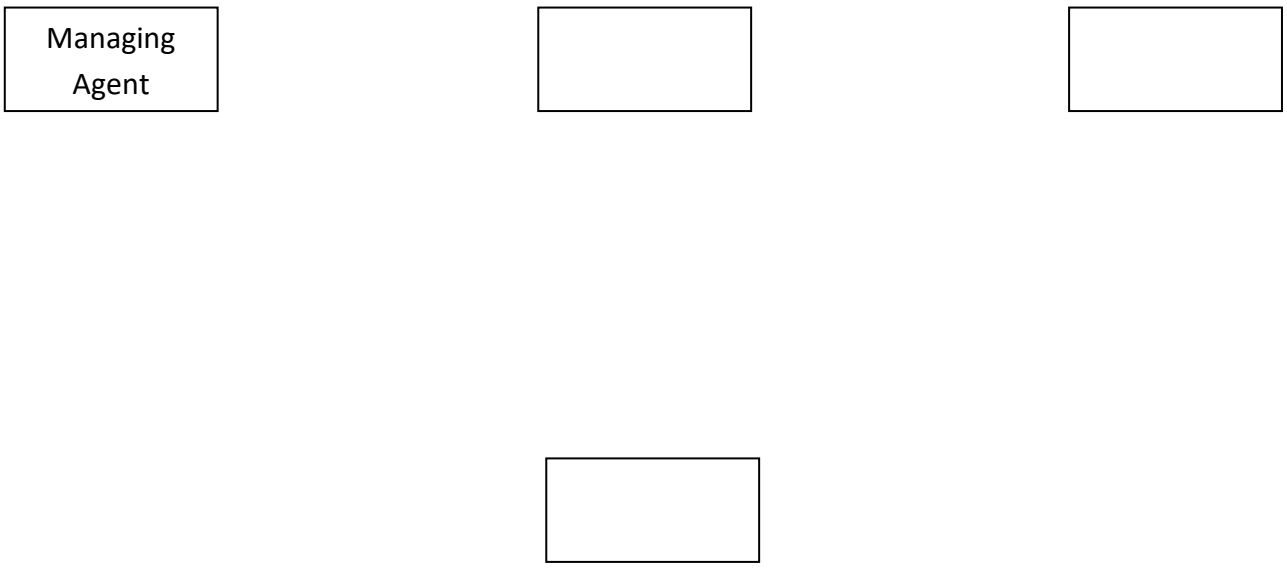


8. An Airline has many Flights, but each Flight is scheduled by only one Airline. A Passenger can make many bookings for Flights, but each Booking is for a specific seat on that Flight. Each Flight has many bookings. A Flight goes to many Airports. Each Airport has many Flights landing and Departing from it.

   Complete the following E-R Diagram based on the above description. Make up your own attributes. You may need to print this page and complete your diagram on paper.



9. In a real estate company each Property is managed by one Managing Agent, but each Managing Agent manages many properties. Each Property can have several owners and an Owner may have many Properties. A Tenant only rents one Property at a time, but each Property may have several Tenants.



Activity Nine

1. The table below records information about books.

BookID	Book Title	Author Surname	Author First Name
1	Gone with the Wind	Mitchell	Margaret
2	The Two Towers	Tolkien	J.R.R.
3	The Fellowship of the	Tolkien	J.R.R.
4	The Return of the King	Tolkien	J.R.R.
5	Wilt	Sharpe	Tom
6	Berlin Game	Deighton	Len
7	Mexico Set	Deighton	Len
8	Spy Hook	Deighton	Len
9	Spy Sinker	Deighton	Len

What normal form is the table in?

Normalise this data to Third Normal Form (3NF).

2. The table below shows mechanical components and the parts they contain.

Component ID	Component	Part 1	Part 2	Part 3	Part 4
E122	Engine	Piston	Valve	Piston Ring	Engine Gasket
C323	Carburettor	Float	Carburettor Gasket	Needle	
A432	Brakes	Brake Pads	Discs		

What normal form is the table in?

Normalise this data to Third Normal Form (3NF)

3. A company keeps the following record of IT equipment in a spreadsheet.

Item ID	Item Name	Purchase Date	Purchase Cost	Supplier ID	Supplier Name	Supplier Phone
1	Razer Gaming Mouse	7/2/2019	$52	1	PLE	93653232
2	Microsoft Keyboard	6/2/2019	$38	1	PLE	93653232
3	Logitech Webcam	8/1/2018	$63	1	PLE	93653232
1	Razer Gaming Mouse	1/1/2020	$55	2	PC Case Gear	94323423
3	Logitech Webcam	1/3/2020	$70	2	PC Case Gear	94323423

What normal form is the table in?

Normalise this data to Third Normal Form (3NF).

Activity Ten

1. Watch the YouTube video https://www.youtube.com/watch?v=QjvjeQquon8.

   Explain the main differences identified in the video between a distributed database and a centralised database.

2. Using the following link, read the description of a data warehouse provided by Oracle Australia: https://www.oracle.com/au/database/what-is-a-data-warehouse/

   What are the elements that a typical data warehouse often includes?

3. After reading the notes on data security, consolidate your knowledge by using the internet to find at least two other secondary sources to explain data security. Record the URL and give a brief description of each of your findings

4. Use the internet to find two examples of products that connect to a database and support ODBC.

5. Use the internet to find two examples of data driven websites.
