SELECT: Order statements 1

GCSE

C C

An estate agent in the north east of England has a database of properties that are currently for sale. An extract from the `Properties` table is below:

PropertyID	Address	City	PropertyType	Bedrooms	ListPrice
Dur032	8 New Elvet	Durham	detached	3	350000
Dur070	2 Sherburn Rd	Durham	terraced	2	200000
New635	6A Salters' Rd	Newcastle Upon Tyne	semi-detached	3	220000
New388	29 Leazes Rd	Newcastle Upon Tyne	terraced	4	480000
Sun112	2 Roker Lane	Sunderland	detached	4	450000

Part A

A buyer is using the agents' website to find all available properties in **Sunderland**. Use the lines of code below to construct a SQL query that would return the following fields for the buyer `Address`, `City`, `PropertyType`, `Bedrooms`, `ListPrice`.

Not all of the statements provided are required to solve the problem

Available items

```
FROM table
```

```
SELECT Address, City, PropertyType, Bedrooms, ListPrice
```

```
WHERE 'City' = 'Sunderland';
```

```
FROM Properties
```

```
WHERE City = 'Sunderland';
```

```
SELECT *
```

Part B

Using the table above, which of the properties in the list below would be returned when the follow query is executed:

```
1 SELECT PropertyID
2 FROM Properties
3 WHERE (Bedrooms = 3 OR Bedrooms = 4) AND ListPrice <= 350000;
```

- Dur032, New388, Sun112
- New635
- Dur032, New635

SELECT query 6

GCSE

Dana has been looking at her app purchases. She wants to redownload an app that she thinks she downloaded in November 2021. An extract from the `AppDownloads` table is below:

DownloadID	App	PurchaseDate	Price
187362	Monopoly Classic	2022-03-01	4.99
273340	Mini Football	2022-03-20	2.79
317333	Stardew Valley	2021-11-19	4.99
387737	Game Dev Tycoon	2021-04-28	4.99
622100	Alien Isolation	2022-02-12	12.99

Choose the option that would complete the query below to help Dana find the game.

```
SELECT *
FROM AppDownloads
WHERE _____ ;
```

The date format for SQL is YYYY-MM-DD, i.e. year, month, then day.

- PurchaseDate >= '2021-11-01' AND <= '2021-11-30'
- PurchaseDate >= '2021-11-01' OR '2021-11-30'
- PurchaseDate >= '2021-11-01' OR PurchaseDate <= '2021-11-30'
- PurchaseDate >= '2021-11-01' AND PurchaseDate <= '2021-11-30'

Clinic: select customer

GCSE A Level

P P P P

Repair & Reform is a muscle therapy clinic that provides support for everyday pain and sports injuries. The company uses a relational database to hold the details of appointments, treatments, customers, and staff.

You can study the table layouts and see some sample data on the Repair & Reform database information page.

A sample of the records from the `appointment` table is shown below.

customer_id	treatment_ref	appointment_date	start_time	staff_id	discount
21	UB_arm	2024-01-02	09:00:00	3	0.25
29	LB_foot	2024-01-02	09:30:00	1	0.2
36	LB_thigh	2024-01-02	11:00:00	3	0

The SQL statement below will return all of the appointments in the database.

Edit the statement so that only the records for the customer whose ID number (`customer_id`) is 4 are displayed.

SQL

```
SELECT treatment_ref, staff_id, appointment_date, start_time
FROM appointment;
```

Enter the **total number of appointments** (the number of records returned) for this particular customer as your answer.

Clinic: select consultation

GCSE A Level

P P P P

Repair & Reform is a muscle therapy clinic that provides support for everyday pain and sports injuries. The company uses a relational database to hold the details of appointments, treatments, customers, and staff.

You can study the table layouts and see some sample data on the Repair & Reform database information page.

A sample of the records from the `customer` table is shown below.

customer_id	first_name	last_name	email	telephone
1	Kohen	Ceyhun	kohen106@example.com	07700900489
2	Vaclovas	Derry	vaclovas167@example.net	07700900158
3	Mar	Sylvia	mar160@example.edu	07700900497

A member of staff wants a list of all of the customers who have not yet had an initial consultation. The `customer` table contains a field — `has_consultation` — that indicates whether or not the initial consultation has taken place.

The SQL statement below currently returns the first name, last name, and email address of **all** of the customers in the database.

Edit the SQL statement so it displays only the records for the customers who have **not** had an initial consultation.

SQL

```
SELECT first_name, last_name, email
FROM customer;
```

Enter the **first name** of the customer that appears **at the top** of the list of results produced by your query.

Clinic: select date 1

GCSE A Level

P P P P

Repair & Reform is a muscle therapy clinic that provides support for everyday pain and sports injuries. The company uses a relational database to hold the details of appointments, treatments, customers, and staff.

You can study the table layouts and see some sample data on the Repair & Reform database information page.

The database currently contains records for appointments made between January 2nd and 31st, 2024 (inclusive). A sample of the data in the `appointment` table is shown below.

customer_id	treatment_ref	appointment_date	start_time	staff_id
21	UB_arm	2024-01-02	09:00:00	3
29	LB_foot	2024-01-02	09:30:00	1
36	LB_thigh	2024-01-02	11:00:00	3

The SQL statement below currently returns the date, start time, customer ID, and treatment reference of all of the appointments stored in the database.

SQL

```
SELECT appointment_date, start_time, customer_id, treatment_ref
FROM appointment;
```

Part A Number of appointments

Edit the SQL statement so that only the appointments that took place on the **18th January 2024** are returned.

What are the **total number of appointments** (i.e., the number of records returned) for this date?

When querying this database, you need to use single quotation marks around a date value. This is the standard in most implementations of SQL.

Part B Order by start time

Edit the SQL statement to order the results so that they appear in start time order, with the earliest time shown first.

What is the **customer ID of the first appointment** on the 18th January 2024 when ordered in this way?

Clinic: select date 2

GCSE C C A Level C C

Repair & Reform is a muscle therapy clinic that provides support for everyday pain and sports injuries. The company uses a relational database to hold the details of appointments, treatments, customers, and staff.

You can study the table layouts and see some sample data on the Repair & Reform database information page.

The database currently contains records for appointments made between January 2nd and 31st, 2024 (inclusive). A sample of the data in the `appointment` table is shown below.

customer_id	treatment_ref	appointment_date	start_time	staff_id
21	UB_arm	2024-01-02	09:00:00	3
29	LB_foot	2024-01-02	09:30:00	1
36	LB_thigh	2024-01-02	11:00:00	3

When run, the SQL statement below returns the appointment date, start time, customer ID, and treatment reference for **all** the appointments stored in the database.

SQL

```sql
SELECT appointment_date, start_time, customer_id, treatment_ref
FROM appointment;
```

Part A Appointments between 8th and 14th January 2024

Edit the SQL statement so that only the appointments that took place between the 8th and 14th of January 2024 (inclusive) are returned.

What are the **total number of appointments** (i.e., the number of records returned) that took place between these dates?

When querying this database, you need to use single quotation marks around a date value. This is the standard in most implementations of SQL.

Part B Order by appointment date and start time

Edit the SQL statement from Part A to order the results so that they appear in appointment date order (oldest dates are shown first). Where there are multiple appointments on the same date, they should appear in order of start time (earliest times are shown first).

The query should still return only the appointments made between the 8th and 14th January 2024 (inclusive) as specified in Part A.

Enter the **customer ID** of the customer that appears **at the top** of the list of results produced by your query.

SELECT query 1

A Level

P P

Vanessa is an artist that creates paintings. Each original painting is displayed in her gallery. She doesn't sell the original work but customers can purchase posters of the paintings. There is a limit to how many posters can be produced for each painting. For example, for the 'Constellations' painting there are only 60 posters available. To keep track of how many posters have been sold for each painting, each poster is numbered.

Painting Name	Poster Limit
Nature disturbed	75
Constellations	60

Vanessa uses a relational database to store information about the sales. This includes details about the paintings and the posters she has produced, her customers, and the sales of the posters. The information is modelled using three entities: `Painting`, `Customer` and `Sale`. Each entity is implemented in the database using a table. The description in standard notation for the tables is as follows:

```
Painting(PaintingId, Name, Description, ProductionYear, PosterLimit)
Customer(CustomerId, LastName, FirstName, Email)
Sale(PaintingId, PosterNumber, CustomerId, SaleDate, Cost)
```

Vanessa wants the details of everyone who purchased any poster that she produced in 2019. The report must display the following fields: `PaintingId`, `PosterNumber`, the customer's `FirstName`, `LastName`, and `Email` address. The report must be sorted in order of customer `LastName`.

Vanessa uses an SQL `SELECT` statement to query the database. Part of the statement below is missing. Identify the missing statement.

SQL

```sql
SELECT Sale.PaintingId, Sale.PosterNumber, Customer.LastName,
Customer.FirstName, Customer.Email
FROM Sale, Customer, Painting
WHERE >>> [missing statement] >>>
AND Painting.PaintingId = Sale.PaintingId
AND Painting.ProductionYear = '2019'
ORDER BY Customer.LastName
```

Clinic: select multiple

GCSE A Level

C C C C

Repair & Reform is a muscle therapy clinic that provides support for everyday pain and sports injuries. The company uses a relational database to hold the details of appointments, treatments, customers, and staff.

You can study the table layouts and see some sample data on the Repair & Reform database information page.

The manager wants to find out how many appointments have been made for their standard knee injury treatment and which staff member carried out each appointment.

To get this information, two tables must be accessed:

- `treatment_ref` will be retrieved from the `appointment` table
- `first_name` and `last_name` of the staff member will be retrieved from the `staff` table

A sample of the records from the `appointment` table is shown below.

customer_id	treatment_ref	appointment_date	start_time	staff_id
21	UB_arm	2024-01-02	09:00:00	3
29	LB_foot	2024-01-02	09:30:00	1
36	LB_thigh	2024-01-02	11:00:00	3

And a sample of the records from the `staff` table.

staff_id	first_name	last_name	email	telephone
1	Sondra	Jorginho	sondra656@example.edu	0770090
2	Deepti	Vesna	deepti273@example.edu	0770090

staff_id	first_name	last_name	email	telephone
3	Mariel	Vappu	mariel487@example.com	07700900

The SQL statement shown below is partially complete. Complete the statement so that the manager gets the information she needs. Your statement should select only appointments for the knee injury treatment with the treatment reference of **LB_knee**.

You will need to join the `appointment` and `staff` tables. The common field between these two tables is `staff_id`, which is the primary key of the `staff` table and the foreign key of the `appointment` table.

SQL

```sql
SELECT appointment.treatment_ref, staff.first_name, staff.last_name
FROM appointment, staff;
```

Part A Number of appointments

What are the **total number of appointments** (the number of records returned) for the knee injury treatment?

Part B Staff member

Enter the **last name** of the staff member that appears **at the top** of the list of results produced by your query.