UPDATE query 3

GCSE

P P

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

The owner of the property at `8 New Elvet` wants to change the list price of their property.

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

Complete the SQL query below using the words provided to update the `ListPrice` of the property at 8 New Elvet, Durham to 330000.

```
[          ] Properties

[          ] ListPrice = 330000

[          ] PropertyID = "Dur032";
```

Items:

UPDATE WHERE INSERT SET

INSERT query 4

GCSE

P P

Below is a table showing app downloads for a customer.

Table `AppDownloads`

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

The customer has recently downloaded another app called `Flow Free`. Use the options below to complete the following query that will insert the new download into the table.

```
[          ] INTO [          ] (DownloadID, App, PurchaseDate, Price)
[          ] (931408, [          ], '2022-01-18', 0.00)[          ]
```

Items:

`;` `VALUES` `AppDownloads` `'Flow Free'` `INSERT`

CREATE TABLE query 2

A Level

C C

Kate is developing an application that will distribute a monthly quiz to her fellow college students. The students will access the quiz via the school portal. For October, the theme of the quiz is Black History Month.

The full list of requirements for her application is below:

- A quiz consists of a number of unique multiple choice questions, displayed in the order specified
- Each question has at least three choices, only one of them is correct
- A question can appear in only one quiz
- Once a student submits a quiz, their attempt is recorded in the database and their answers can't be changed
- A student can attempt a quiz more than once

The table definitions for the database are:

```
Quiz(QuizId, Name, PublishedDate)
Question(QuestionId, QuizId, Description, QuestionOrder)
Choice(ChoiceId, QuestionId, Description, ChoiceOrder, IsCorrect)
Student(StudentId, FirstName, LastName, YearGroup, Email)
Attempt(AttemptId, StudentId, QuizId, SubmittedDate, SubmittedTime)
AttemptAnswer(AttemptId, ChoiceId)
```

Put the following partial SQL statements in order to create the `Question` table in the database. You only need to use four partial statements.

Available items

```
  PRIMARY KEY (QuestionId));
```

```
  Description VARCHAR(100),
  QuestionOrder INTEGER,
```

```
  QuestionId INTEGER,
  QuizId INTEGER,
```

```
  Description VARCHAR(100),
  QuestionOrder BOOLEAN,
```

```
  PRIMARY KEY (QuestionId)
  FOREIGN KEY (QuizId) REFERENCES Quiz(QuizId));
```

```
CREATE TABLE Question (
```

INSERT query 3

A Level

C C

Kate is developing an application that will distribute a monthly quiz to her fellow college students. The students will access the quiz via the school portal. For October, the theme of the quiz is Black History Month.

The full list of requirements for her application is below:

- A quiz consists of a number of unique multiple choice questions
- Each question has at least three choices, only one of them is correct
- A question can appear in only one quiz
- Once a student submits a quiz, their attempt is recorded in the database and their answers can't be changed
- A student can attempt a quiz more than once

The table definitions for the database are:

```
Quiz(QuizId, Name, PublishedDate)
Question(QuestionId, QuizId, Description, QuestionOrder)
Choice(ChoiceId, QuestionId, Description, ChoiceOrder, IsCorrect)
Student(StudentId, FirstName, LastName, YearGroup, Email)
Attempt(AttemptId, StudentId, QuizId, SubmittedDate, SubmittedTime)
AttemptAnswer(AttemptId, ChoiceId)
```

Kate wants to create the records for the Black History Month quiz in the database. So far she has come up with the quiz name and the description for the first question:

Quiz Name: Black History Month
Question 1: What is Mae Jemison famous for?

Select the correct partial SQL statements and drag them into the correct order to create the records for the quiz and question as above. You only need to use six partial statements.

Available items

```
(QuestionId, QuizId, Description, QuestionOrder)
```

```
INSERT INTO Question
```

```
VALUES (5,'Black History Month',01/10/2021);
```

```
(QuizId, Name, PublishedDate)
```

```
VALUES (5,'Black History Month','01/10/2021');
```

```
VALUES (1,5,'What is Mae Jemison famous for?','Q1');
```

```
VALUES (1,5,'What is Mae Jemison famous for?',1);
```

```
INSERT INTO Quiz
```

INSERT query 2

A Level

C C

Suhreena wants to create a program for her A level project. The program will allow her to run polls that the students can participate in using the school portal.

- Each poll consists of one question and any number of response options. Each response option belongs to a specific poll.
- Each student can only vote for one option in each poll. Once a vote is submitted, it can't be changed.
- When a poll is over, it can be set as inactive.

Suhreena wants to use a relational database to store information about the polls. The information is modelled using four entities `Student`, `Poll`, `Option` and `Vote`. Each entity is implemented in the database using a table. The description in standard notation for the entities is as follows:

```
Student(StudentId, FirstName, LastName, YearGroup)
Poll(PollId, QuestionText, IsActive, PublishedDate)
Option(OptionId, PollId, OptionText)
Vote(StudentId, OptionId, SubmittedDate, SubmittedTime)
```

Suhreena wants to run a poll among the Y12 and Y13 students about the impact of climate change.

- The poll question is: **Which is a higher risk of climate change?**.
- The poll options are: **extreme temperatures** and **sea level rise**.

The fields `Poll.PollId` and `Option.PollId` form a primary/foreign key relationship between the `Poll` and `Option` tables. Therefore, referential integrity is enforced when adding records to these tables.

Select the correct SQL statements and drag them into the correct order so that the records for the poll question and poll options are created in the database. You only need to use **three** statements.

Available items

```
INSERT INTO Poll (PollId, QuestionText, IsActive, PublishedDate)
VALUES (1, 'Which is a higher risk of climate change?', TRUE, '01/10/2021')
```

```
INSERT INTO Option (OptionId, PollId, OptionText)
VALUES (1, 2, 'sea level rise')
```

```
INSERT INTO Option (OptionId, PollId, OptionText)
VALUES (1, 1, 'extreme temperatures')
```

```
INSERT INTO Option (OptionId, PollId, OptionText)
VALUES (2, 1, 'sea level rise')
```

Clinic: delete 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 `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

The customer whose ID is 2 has moved away and will no longer be able to attend their scheduled appointments. These appointments must be deleted from the database.

The SQL statement shown below has a placeholder. You must replace this with a statement that will delete all the records from the `appointment` table for the relevant customer. Below the placeholder is a `SELECT` query that counts the total number of appointments. **Do not delete the SELECT statement** as you need this to answer the question.

Reload the page to restore the original state of the question and the database if you accidentally delete the wrong data.

SQL

```
1  -- Write delete statement here --
2
3
4  SELECT COUNT(*) AS total_appointments
```

Enter the **total number of appointments** that remain in the database once the appointments for the customer with ID number 2 have been deleted.

Clinic: delete appointments

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

One of the consultation rooms will be out of use on the 11th of January 2024 while some maintenance work is carried out. The room is used for appointments that have one of the following treatment references stored in the `treatment_ref` field: `C_first`, `C_further`, or `C_plan`.

All appointments on this day have been rearranged and the existing appointment records must be deleted.

The SQL statement shown below has a placeholder. You must replace this with a statement that will delete all the records from the `appointment` table for the relevant date and treatment references. Below the placeholder is a `SELECT` query that counts the total number of appointments. **Do not delete the SELECT statement** as you need this to answer the question.

Reload the page to restore the original state of the question and the database if you accidentally delete the wrong data.

SQL

```
1  -- Write delete statement here --
2
3
4  SELECT COUNT(*) AS total_appointments
```

Enter the **total number of appointments** remaining once the relevant appointments have been deleted.

Clinic: update price

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 `treatment` table is shown below.

treatment_ref	treatment_name	type	description	duration	price
C_first	First consultation	consultation	An initial consultation...	60	50
LB_foot	Foot & ankle treatment	deep tissue	Treatment that can...	30	30
UB_back	Back injury	trigger point	Relieves the tissues...	60	45

The manager wants to reduce the price of the back injury treatment from £45 to £40. The reference (`treatment_ref`) of this treatment is `UB_back`.

The SQL statement shown below has a placeholder. You need to replace this with an `UPDATE` statement that updates the price of the relevant treatment to £40. The price is stored as a number so you must not enter the pound sign (£).

Below the placeholder is a `SELECT` statement that calculates the average price of a treatment. **Do not delete this SELECT statement** as you need to run this to answer the question.

Reload the page to restore the original state of the question and the database if you accidentally update the wrong data.

SQL

```
1  -- Write update statement here --
2
3
4  SELECT AVG(price)
```

Enter the **average price** of all the treatments once the price of the back injury treatment has been updated.

Clinic: insert treatment

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 `treatment` table is shown below.

treatment_ref	treatment_name	type	description	duration	price
C_first	First consultation	consultation	An initial consultation...	60	50
LB_foot	Foot & ankle treatment	deep tissue	Treatment that can...	30	30
UB_back	Back injury	trigger point	Relieves the tissues...	60	45

The manager wants to add a new treatment for hand and wrist injuries. The information for the new treatment is shown below for each of the fields.

Field name	Data type	Data to insert
`treatment_ref`	Text	UB_hand
`treatment_name`	Text	Hand and wrist injury
`type`	Text	deep tissue
`description`	Text	To relieve muscle and tendon pain in the hand and wrist.
`duration`	Integer	20
`price`	Real	27.5

The SQL statement shown below has a placeholder. You need to replace this with a statement to `INSERT` the new treatment record with all of the information that is needed (as specified above).

Below the placeholder is a `SELECT` statement that calculates the average duration of a treatment. **Do not delete this SELECT statement** as you need to run this to answer the question.

Reload the page to restore the original state of the question and the database if you accidentally update the wrong data.

SQL

```sql
-- Write insert statement here --


SELECT AVG(duration)
```

Enter the **average duration** of a treatment once the new hand and wrist injury treatment has been inserted.
