

Q1. Explain why the following table does not qualify as a relation and describe a potential solution to fix the issue.

SID	Student Name	Student Contact	City
87292	Mark Anthony	+61788292342 ; +61282662834	Sydney
91623	Tim Berrel	+61639715239 ; +61427882639	Sydney
64872	Violet Aura	+61912213572 ; +61732215543	Sydney
42881	Karl Brown	+61528193732 ; +61633037253	Sydney

Q2. Consider the following tables below. Write the SQL query to:

a. Create both the tables.

Movie Details Table:

Movie_ID	Movie_Title	Year_of_Release
1001	The Avengers: Age of Ultron	2015
1002	Harry Potter and the Sorcerer's Stone	2001
1003	Percy Jackson & the Olympians: The lightning Thief	2010

Sales Order Table:

Order_ID	Date_Of_Purchase	Store_ID	Movie_ID
0001	11/11/2018	9403AF	1001
0002	10/02/2019	9403AF	1003
0003	25/06/2016	4572ZH	1003
0004	04/12/2023	9403AF	1002
0005	29/07/2020	4572ZH	1002
0006	13/01/2024	4572ZH	1003

Q3. You are hired as a data modelling expert by Star Truck Suppliers. Given the immense amount of data, how would you decide whether to use a star schema or snowflake schema to use for data modelling.

Q4.

Consider the following Orders and Customers tables:

Customers Table:

customer_id	customer_name	email
1	John Doe	john@example.com
2	Jane Smith	jane@example.com
3	Bob Brown	bob@example.com

Orders Table:

order_id	customer_id	product_name	order_date
101	1	Laptop	2024-07-12
102	2	Smartphone	2024-07-15
103	1	Tablet	2024-07-18
104	3	Headphones	2024-07-20

1. Identify the primary key in both tables. Explain why you selected those keys.
2. What is the difference between a primary key and a foreign key?
3. Write an SQL command to create these tables

Q5.

You have two tables, employees and departments, in your database. The employees table contains the following columns: employee_id, first_name, last_name, department_id, and salary. The departments table contains the columns: department_id and department_name. Write a SQL query to retrieve the first_name of each employee along with the department_name they belong to.

Sample Solutions:

Q1.

The following table does not qualify as a relation because the Student Contact attribute is not an atomic attribute. **Atomic attributes** mean that the attribute can only contain a single value. Since each row in the Student Contact column has more than one value, the Student Contact column is not an atomic attribute, rather it is a composite or multivalued attribute. One potential solution to fix this issue is to create another relation/table that maps a student's ID to that student's contact number.

Q2.

```
CREATE TABLE Movie_Details (  
    Movie_ID INT PRIMARY KEY,  
    Movie_Title VARCHAR(255),  
    Year_of_Release INT  
);
```

#Code to create Movie Details Table

```
CREATE TABLE Sales_Order (  
    Order_ID CHAR(4) PRIMARY KEY,  
    Date_Of_Purchase DATE,  
    Store_ID VARCHAR(10),  
    Movie_ID INT,  
    FOREIGN KEY (Movie_ID) REFERENCES Movie_Details(Movie_ID)  
);
```

Q3.

The decision to either implement a snowflake schema or a star schema really depends on the business's needs. Typically, a snowflake schema is used when you want to store data more efficiently and perform detailed analyses on the dataset. This type of schema typically requires a lot of joins which may impact the query processing speed. On the other hand, if the business requires fast query processing speeds and does not have a very complex dataset, then the star schema is the optimal choice. Star Schema has Higher redundancy in dimension tables because of denormalization. Snowflake Schema has lower redundancy due to normalization, which avoids repeated data.

Q4.

Part1:

Customers Table:

- **Primary Key: customer_id**

Explanation:

The customer_id is chosen as the primary key because it uniquely identifies each customer in the Customers table. No two customers will have the same customer_id, making it an appropriate choice for a primary key. It ensures that each customer record is unique and can be referenced accurately.

Orders Table:

- **Primary Key: order_id**

Explanation:

The order_id is chosen as the primary key because it uniquely identifies each order in the Orders table. Each order has a distinct order_id, ensuring that no two orders are the same. This makes order_id the appropriate choice for a primary key, allowing for precise tracking and management of orders.

Part2:

A **Primary Key** uniquely identifies each record within a table and must be unique and non-null. There can be up to one per table. In contrast, a **Foreign Key** establishes a relationship between tables by referring to a primary key in another table; it can contain duplicate values and NULLs. The primary key ensures entity integrity within its own table, while the foreign key maintains referential integrity between related tables.

Part3:

```
CREATE TABLE Customers (  
    customer_id INT PRIMARY KEY,  
    customer_name VARCHAR(100) NOT NULL,  
    email VARCHAR(100) NOT NULL UNIQUE  
);  
CREATE TABLE Orders (  
    order_id INT PRIMARY KEY,  
    customer_id INT,  
    product_name VARCHAR(100) NOT NULL,  
    order_date DATE NOT NULL,  
    FOREIGN KEY (customer_id) REFERENCES Customers(customer_id)  
);
```

Q5.

```
SELECT employees.first_name, departments.department_name  
FROM employees  
INNER JOIN departments ON employees.department_id = departments.department_id;
```