1. Create a new table without using CREATE statement, with the same schema of an existing table without any data. It should be a single statement.

**Sample Input:**

Base Table

| **ID** | **Name** |
| --- | --- |
| 1 | Ram |
| 2 | Sai |
| 3 | Ali |
| 4 | Venkat |
| 5 | Reinhard |

**Sample Output:**

New Table

| **ID** | **Name** |
| --- | --- |

**Script – DDL and TINSERT Sample Data**

| CREATE TABLE BaseTable  (   [ID] INT     ,[Name] VARCHAR(10)  );    INSERT INTO BaseTable VALUES  (1,'Ram'),  (2,'Sai'),  (3,'Ali'),  (4,'Venkat'),  (5,'Reinhard');    SELECT \* FROM BaseTable |
| --- |

2. We need to provide rank. The logic for the ranking is that whenever you have a value “Product” then you must start the new number and all others below should follow the same rank. Please check out the sample input and the expected output.

**Sample Input:**

| **ID** | **Vals** |
| --- | --- |
| 1 | Product |
| 2 | Milk |
| 3 | Butter |
| 4 | Cheese |
| 5 | Yogurt |
| 6 | Product |
| 7 | Muesli |
| 8 | Porridge |
| 9 | Product |
| 10 | Banana |

**Sample Output:**

| **ID** | **Vals** | **Rank** |
| --- | --- | --- |
| 1 | Product | 1 |
| 2 | Milk | 1 |
| 3 | Butter | 1 |
| 4 | Cheese | 1 |
| 5 | Yogurt | 1 |
| 6 | Product | 2 |
| 7 | Muesli | 2 |
| 8 | Porridge | 2 |
| 9 | Product | 3 |
| 10 | Banana | 3 |

**Script – DDL and INSERT Sample Data**

| CREATE TABLE RankingPuzzle  (       ID INT      ,Vals VARCHAR(10)  );    INSERT INTO RankingPuzzle VALUES  (1,'Product'),  (2,'Milk'),  (3,'Butter'),  (4,'Cheese'),  (5,'Yogurt'),  (6,'Product'),  (7,'Muesli'),  (8,'Porridge'),  (9,'Product'),  (10,'Banana');    SELECT \* FROM RankingPuzzle |
| --- |

3. Need to find the sum of sales for each employee. We need to follow the following conditions while summing up.

1. for inactive employees (IsActive=0) group the sum under Inactive name and rest under their respective names.

b. need to do this in a **single select.**

Please check out the sample input and the expected output.

**Sample Input:**

| **EmpName** | **SaleAmount** | **IsActive** |
| --- | --- | --- |
| Pawan | 2500 | 1 |
| Pawan | 3000 | 1 |
| Avtaar | 800 | 1 |
| Avtaar | 1000 | 1 |
| Kishan | 2800 | 1 |
| Kishan | 3000 | 1 |
| Nimit | 500 | 1 |
| Nimit | 800 | 1 |
| Kavita | 1000 | 0 |
| Lakshmi | 1000 | 0 |
| Madhu | 500 | 0 |
| Joyeeta | 2500 | 0 |

**Sample Output:**

| **EmpName** | **SaleAmount** |
| --- | --- |
| Avtaar | 1800 |
| Inactive | 5000 |
| Kishan | 5800 |
| Nimit | 1300 |
| Pawan | 5500 |

**Script – DDL and INSERT Sample Data**

| CREATE TABLE SalesData  (      EmpName VARCHAR(256),      SaleAmount DECIMAL(18,2),      IsActive BIT  );    INSERT INTO SalesData VALUES  ('Pawan',2500.00,1),  ('Pawan',3000.00,1),  ('Avtaar',800.00,1),  ('Avtaar',1000.00,1),  ('Kishan',2800.00,1),  ('Kishan',3000.00,1),  ('Nimit',500.00,1),  ('Nimit',800.00,1),  ('Kavita',1000.00,0),  ('Lakshmi',1000.00,0),  ('Madhu',500.00,0),  ('Joyeeta',2500.00,0);    SELECT \* FROM SalesData |
| --- |

4. Refer to below Stage and Mart tables. If a particular employee is no longer present in the stage table then update the IsValid flag to No for that particular employee in the Mart table. **Note: Don’t use a Subquery.**

**Stage Table:**

| **EmpID** | **Name** |
| --- | --- |
| 1 | Kiran |
| 2 | Vikram |
| 3 | Chaitra |

**Mart Table:**

| **EmpID** | **Name** | **IsValid** |
| --- | --- | --- |
| 1 | Kiran | Yes |
| 2 | Ruthvik | Yes |
| 3 | Vikram | Yes |
| 4 | Chaitra | Yes |
| 5 | Vimal | Yes |

**Output:**

| **EmpID** | **Name** | **IsValid** |
| --- | --- | --- |
| 1 | Kiran | Yes |
| 2 | Ruthvik | No |
| 3 | Vikram | Yes |
| 4 | Chaitra | Yes |
| 5 | Vimal | No |

**Script – DDL and INSERT Sample Data**

| --Create Stage table  CREATE TABLE stagetable  (  EmpID INT  ,Name VARCHAR(100)  )  --Insert Data  INSERT INTO stagetable  VALUES  (1,'Kiran'),  (2,'Vikram'),  (3,'Chaitra')    --Create Mart table  CREATE TABLE marttable  (  EmpID INT  ,Name VARCHAR(100)  ,IsValid NVARCHAR(10)  )  --Insert Data  INSERT INTO marttable  VALUES  (1,'Kiran','Yes'),  (2,'Ruthvik','Yes'),  (3,'Vikram','Yes'),  (4,'Chaitra','Yes'),  (5,'Vimal','Yes') |
| --- |

5. Write a query to verify whether a particular column is the grain column or primary key of a particular table using a single select statement and without using distinct keyword