

Task 1: Create Table Queries

```
-- Create staging table
CREATE TABLE Staging (
  Customer_Name VARCHAR(255),
  Customer_ID VARCHAR(18),
  Open_Date DATE,
  Last_Consulted_Date DATE,
  Vaccination_Type CHAR(5),
  Doctor_Consulted VARCHAR(255),
  State CHAR(5),
  Country CHAR(5),
  Post_Code INT,
  DOB DATE,
  Active_Customer CHAR(1)
);

-- Create country-specific table (e.g India)
CREATE TABLE Table_India (
  Customer_Name VARCHAR(255),
  Customer_ID VARCHAR(18),
  Open_Date DATE,
  Last_Consulted_Date DATE,
  Vaccination_Type CHAR(5),
  Doctor_Consulted VARCHAR(255),
  State CHAR(5),
  Country CHAR(5),
  Post_Code INT,
  DOB DATE,
  Active_Customer CHAR(1),
  Age INT,
  Days_Since_Last_Consulted INT
);
```

Task 2: Create Tables with Derived Columns

```
-- Create the table with additional derived columns
CREATE TABLE Table_India (
  Customer_Name VARCHAR(255),
  Customer_ID VARCHAR(18),
  Open_Date DATE,
  Last_Consulted_Date DATE,
  Vaccination_Type CHAR(5),
```

```

    Doctor_Consulted VARCHAR(255),
    State CHAR(5),
    Country CHAR(5),
    Post_Code INT,
    DOB DATE,
    Active_Customer CHAR(1),
    Age INT,
    Days_Since_Last_Consulted INT
);

-- Insert data with the derived columns and filter where last consulted > 30 days
INSERT INTO Table_India
SELECT
    Customer_Name,
    Customer_ID,
    Open_Date,
    Last_Consulted_Date,
    Vaccination_Type,
    Doctor_Consulted,
    State,
    Country,
    Post_Code,
    DOB,
    Active_Customer,
    FLOOR(DATEDIFF(CURRENT_DATE, DOB) / 365) AS Age,
    DATEDIFF(CURRENT_DATE, Last_Consulted_Date) AS Days_Since_Last_Consulted
FROM Staging
WHERE Country = 'IND'
AND DATEDIFF(CURRENT_DATE, Last_Consulted_Date) > 30;

```

Task 3: Create necessary validations

```

-- 1. Validate if DOB is in the future
SELECT * FROM Staging WHERE DOB > CURRENT_DATE;

-- 2. Validate if Last_Consulted_Date is in the future
SELECT * FROM Staging WHERE Last_Consulted_Date > CURRENT_DATE;

-- 3. Validate if Last_Consulted_Date is before Open_Date
SELECT * FROM Staging WHERE Last_Consulted_Date < Open_Date;

-- 4. Validate if Post_Code contains non-numeric values
SELECT * FROM Staging WHERE TRY_CAST(Post_Code AS INT) IS NULL;

```

```
-- 5. Validate if Active_Customer contains invalid values
SELECT DISTINCT Active_Customer FROM Staging WHERE Active_Customer NOT IN ('Y',
'N');
```