Abstract :

The Smart Farming Systematic Approach Data Management System aims to assist farmers by providing suggestions for systematic farming. This project employs the use of data management systems and technologies to analyze various aspects of farming such as soil quality, weather patterns, and crop productivity. By collecting and analyzing this data, the system provides recommendations to farmers on how to optimize their farming practices to improve yields and reduce costs. Through the implementation of this project, farmers can make informed decisions and improve their overall agricultural productivity, ultimately leading to increased profitability and sustainability in the farming industry.

****

Tables :

1. Farmers table - containing information on the farmers, including their name, contact details, and location.

2. Farms table - containing information on the farms, including the size of the farm, the crops grown, and the soil quality.

3. Weather table - containing information on the weather patterns in the area, including temperature, humidity, and rainfall.

4. Crop table - containing information on the crops, including the crop type, expected yield, and harvest time.

5. Recommendations table - containing recommendations for farmers based on the data analysis performed by the system, including suggestions on crop selection, planting time, and fertilization schedules.

****

Cardinalities and constraints :

1. Farmers table:
   - Mapping : One farmer can have many farms.
   - Participation constraint: Each farm must be associated with a farmer.
   - Key constraint: Farmer ID is the primary key for this table.

2. Farms table:
   - Mapping : One farm belongs to one farmer.
   - Participation constraint: Each farm must have a farmer associated with it.
   - Key constraint: Farm ID is the primary key for this table.

3. Weather table:
   - Mapping : One weather record is associated with one farm.
   - Participation constraint: Each farm must have a weather record associated with it.
   - Key constraint: Weather ID is the primary key for this table.

4. Crop table:
   - Mapping : One crop is grown on many farms.
   - Participation constraint: Each farm must have at least one crop associated with it.
   - Key constraint: Crop ID is the primary key for this table.

5. Recommendations table:
   - Mapping : One recommendation is associated with one farm.
   - Participation constraint: Each farm must have at least one recommendation associated with it.
   - Key constraint: Recommendation ID is the primary key for this table.

Additionally, the following key constraints are also present :

   - Farmer ID in the Farmers table is a foreign key that references the primary key in the Farms table.
   - Farm ID in the Farms table is a foreign key that references the primary key in the Weather table.
   - Farm ID in the Farms table is a foreign key that references the primary key in the Crop table.
   - Farm ID in the Farms table is a foreign key that references the primary key in the Recommendations table.

****

DDL Operations :

```
CREATE TABLE Farmers (
  Farmer_ID INT PRIMARY KEY,
  Name VARCHAR(255) NOT NULL,
  Contact VARCHAR(255) NOT NULL,
  Location VARCHAR(255) NOT NULL
);
```

output:
Query OK, 0 rows affected

```
CREATE TABLE Farms (
  Farm_ID INT PRIMARY KEY,
```

```sql
  Farmer_ID INT NOT NULL,
  Size FLOAT NOT NULL,
  Crop VARCHAR(255) NOT NULL,
  Soil_Qual VARCHAR(255) NOT NULL,
  FOREIGN KEY (Farmer_ID) REFERENCES Farmers(Farmer_ID)
);
```

output:
Query OK, 0 rows affected

```sql
CREATE TABLE Weather (
  Weather_ID INT PRIMARY KEY,
  Farm_ID INT NOT NULL,
  Temp FLOAT NOT NULL,
  Humidity FLOAT NOT NULL,
  Rainfall FLOAT NOT NULL,
  FOREIGN KEY (Farm_ID) REFERENCES Farms(Farm_ID)
);
```

output:
Query OK, 0 rows affected

```sql
CREATE TABLE Crop (
  Crop_ID INT PRIMARY KEY,
  Type VARCHAR(255) NOT NULL,
  Yield FLOAT NOT NULL,
  Harvest DATE NOT NULL,
  Farm_ID INT NOT NULL,
  FOREIGN KEY (Farm_ID) REFERENCES Farms(Farm_ID)
);
```

output:
Query OK, 0 rows affected

```sql
CREATE TABLE Recommendations (
  RecommendationID INT PRIMARY KEY,
  Farm_ID INT NOT NULL,
  Details VARCHAR(255) NOT NULL,
  FOREIGN KEY (Farm_ID) REFERENCES Farms(Farm_ID)
);
```

output:
Query OK, 0 rows affected

DML Operations :

INSERT INTO Farmers (Farmer_ID, Name, Contact, Location)
VALUES
  (1, 'John Smith', '123-456-7890', 'California'),
  (2, 'Jane Doe', '555-555-5555', 'Florida'),
  (3, 'Bob Johnson', '555-123-4567', 'Texas'),
  (4, 'Sarah Lee', '888-888-8888', 'New York'),
  (5, 'Tom Brown', '777-777-7777', 'Arizona');

output:
Query OK, 5 rows affected

INSERT INTO Farms (Farm_ID, Farmer_ID, Size, Crop, Soil_Qual)
VALUES
  (1, 1, 10.0, 'Corn', 'Loamy'),
  (2, 1, 5.0, 'Soybeans', 'Sandy'),
  (3, 2, 20.0, 'Wheat', 'Clay'),
  (4, 3, 15.0, 'Barley', 'Loamy'),
  (5, 4, 8.0, 'Potatoes', 'Sandy');

output:
Query OK, 5 rows affected

INSERT INTO Weather (Weather_ID, Farm_ID, Temp, Humidity, Rainfall)
VALUES
  (1, 1, 75.0, 50.0, 0.2),
  (2, 2, 85.0, 65.0, 0.0),
  (3, 3, 65.0, 70.0, 0.5),
  (4, 4, 60.0, 45.0, 0.1),
  (5, 5, 70.0, 60.0, 0.3);

output:
Query OK, 5 rows affected

INSERT INTO Crop (Crop_ID, Type, Yield, Harvest, Farm_ID)
VALUES
  (1, 'Corn', 100.0, '2022-10-15', 1),
  (2, 'Soybeans', 75.0, '2022-09-30', 2),
  (3, 'Wheat', 60.0, '2022-06-01', 3),
  (4, 'Barley', 80.0, '2022-07-15', 4),
  (5, 'Potatoes', 90.0, '2022-11-01', 5);

output:
Query OK, 5 rows affected

```sql
INSERT INTO Recommendations (RecommendationID, Farm_ID, Details)
VALUES
  (1, 1, 'Apply nitrogen fertilizer'),
  (2, 2, 'Rotate crops to prevent disease'),
  (3, 3, 'Use cover crops to improve soil quality'),
  (4, 4, 'Plant resistant varieties to combat pests'),
  (5, 5, 'Increase soil moisture for better yields');
```

output:
Query OK, 5 rows affected