tables

**user** = admin, farmer, company, service provider {Id Name/company name contact email(optional) address password typeOfUser}

**property** = farmers can add the property details in it {property id , farmerid, location, type of land(bagayti or jirayati), img of land, img of 7/12 , area in acre, area in guntha}

**application** = company can apply for the property {application id, company id, property id, status(accepted or not)}

**service** = service provider can provide the service {service id, serviceprovider id, name , service name , price, description}

**service** \_applications = provide the applications for the service {application id, service id, company id, status(accpetd/reject), iscomplete}

**contact** = all the contact us informations {contact id, name, email(optional), Description}

**payment** = 1000 registration fee up to register the farm with the farmer {payment id, service id/property id, payment method, payment mode, payment Status}

**governmentSchems** = admin adds the farming related schems on the portal {scheme id, title, price, start date, last date, description}  
  
  
**Table Relation:**

1. **User => Property (One-to-Many):** One farmer can add multiple properties.
2. **User => Application (One-to-Many):** One company can apply for multiple properties.
3. **User => Service (One-to-Many):** One service provider can offer multiple services.
4. **User => Service Applications (One-to-Many):** One company can apply for multiple services.
5. **Property => Application (One-to-Many):** One property can receive multiple applications.
6. **Property => Payment (One-to-One):** Each property has one registration payment.
7. **Service => Service Applications (One-to-Many):** One service can have multiple applications.
8. **Service => Payment (One-to-One):** A service payment can be tracked.
9. **Application => (User, Property) (Many-to-One):** A company applies for a property.
10. **Service Applications => (User, Service) (Many-to-One):** A company applies for a service.
11. **Government Schemes:** Independent (Managed by Admin).

**Creating Tables**

**1.Creating User Table**

Create a table for the users which includes farmers, companies, ServiceProviders, admin

CREATE TABLE finaldb.user (

Id INT PRIMARY KEY AUTO\_INCREMENT,

Name VARCHAR(255) NOT NULL,

Contact VARCHAR(20) NOT NULL UNIQUE,

Email VARCHAR(255) UNIQUE,

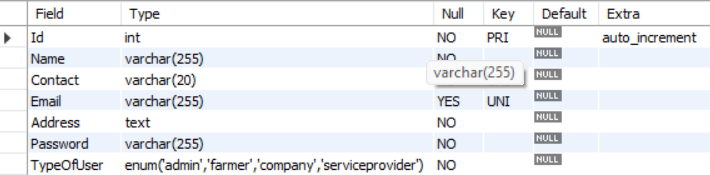
Address TEXT NOT NULL,

Password VARCHAR(255) NOT NULL,

TypeOfUser ENUM('admin', 'farmer', 'company', 'serviceprovider') NOT NULL

);

**Design:**



**2.Create the table property:**

**Create a table where farmers can add there properties details.**

CREATE TABLE finaldb.property (

PropertyId INT PRIMARY KEY AUTO\_INCREMENT,

FarmerId INT NOT NULL,

Location TEXT NOT NULL,

TypeOfLand ENUM('bagayti', 'jirayati') NOT NULL,

ImgOfLand VARCHAR(255) NOT NULL,

ImgOf7\_12 VARCHAR(255) NOT NULL,

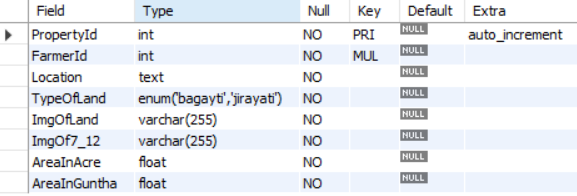
AreaInAcre FLOAT NOT NULL CHECK (AreaInAcre > 0),

AreaInGuntha FLOAT NOT NULL CHECK (AreaInGuntha > 0),

FOREIGN KEY (FarmerId) REFERENCES finaldb.user(Id) ON DELETE CASCADE

);

**Design:**



**3.Create table for services**

**Create a table where service provider can add there services**

CREATE TABLE finaldb.service (

ServiceId INT PRIMARY KEY AUTO\_INCREMENT,

ServiceProviderId INT NOT NULL,

Name VARCHAR(255) NOT NULL,

ServiceName VARCHAR(255) NOT NULL,

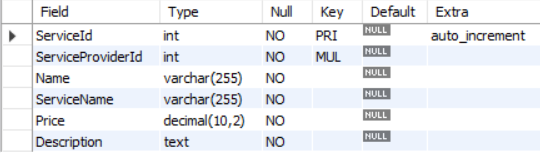
Price DECIMAL(10,2) NOT NULL CHECK (Price >= 0),

Description TEXT NOT NULL,

FOREIGN KEY (ServiceProviderId) REFERENCES finaldb.user(Id) ON DELETE CASCADE

);

**Design:**

****

**4.Create the table for applications of property:**

**Create a table where company can apply for they are interested in the property**

CREATE TABLE finaldb.application (

ApplicationId INT PRIMARY KEY AUTO\_INCREMENT,

CompanyId INT NOT NULL,

PropertyId INT NOT NULL,

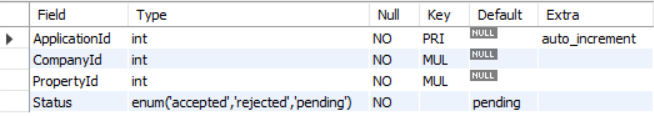
Status ENUM('accepted', 'rejected', 'pending') NOT NULL DEFAULT 'pending',

FOREIGN KEY (CompanyId) REFERENCES finaldb.user(Id) ON DELETE CASCADE,

FOREIGN KEY (PropertyId) REFERENCES finaldb.property(PropertyId) ON DELETE CASCADE

);

**Design:**

****

**5.Create the table for application of services:**

Create a table where company or farmers can apply for the service

CREATE TABLE finaldb.service\_applications (

ApplicationId INT PRIMARY KEY AUTO\_INCREMENT,

ServiceId INT NOT NULL,

CompanyId INT NOT NULL,

Status ENUM('accepted', 'rejected', 'pending') NOT NULL DEFAULT 'pending',

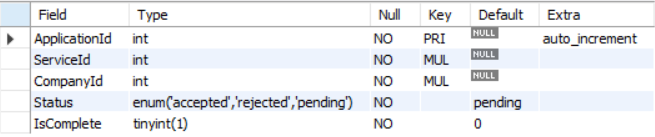
IsComplete BOOLEAN NOT NULL DEFAULT FALSE,

FOREIGN KEY (ServiceId) REFERENCES finaldb.service(ServiceId) ON DELETE CASCADE,

FOREIGN KEY (CompanyId) REFERENCES finaldb.user(Id) ON DELETE CASCADE

);

Design:



**6.Create table for the contact us**

Create a table for the contact us page

CREATE TABLE finaldb.contact (

ContactId INT PRIMARY KEY AUTO\_INCREMENT,

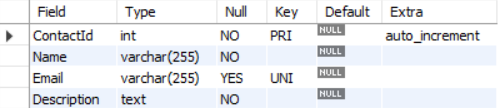
Name VARCHAR(255) NOT NULL,

Email VARCHAR(255) UNIQUE,

Description TEXT NOT NULL

);

**Design:**



**7.Create table for payment details:**

Create a table for the payment details

CREATE TABLE finaldb.payment (

PaymentId INT PRIMARY KEY AUTO\_INCREMENT,

ServiceOrPropertyId INT NOT NULL,

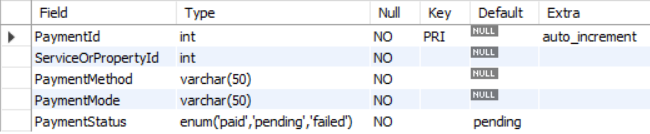
PaymentMethod VARCHAR(50) NOT NULL,

PaymentMode VARCHAR(50) NOT NULL,

PaymentStatus ENUM('paid', 'pending', 'failed') NOT NULL DEFAULT 'pending'

);

**Design:**



**8.Create table for the government Schemas:**

**Create a table where admin can add the government Schemas**

CREATE TABLE finaldb.governmentSchemes (

SchemeId INT PRIMARY KEY AUTO\_INCREMENT,

Title VARCHAR(255) NOT NULL UNIQUE,

Price DECIMAL(10,2) NOT NULL CHECK (Price >= 0),

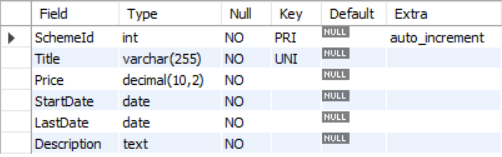
StartDate DATE NOT NULL,

LastDate DATE NOT NULL,

Description TEXT NOT NULL

);

**Design:**



**Inserting Sample Data**

**1.Inserting data in User Table:**

INSERT INTO user (Name, Contact, Email, Address, Password, TypeOfUser) VALUES

('Admin User', '1234567890', 'admin@example.com', 'Admin Address', 'adminpass', 'admin'),

('Farmer1', '9876543210', 'farmer1@example.com', 'Farmer Address', 'farmerpass', 'farmer'),

('Company1', '1122334455', 'company1@example.com', 'Company Address', 'companypass', 'company'),

('ServiceProvider1', '6677889900', 'sp1@example.com', 'SP Address', 'spass', 'serviceprovider'),

('Farmer2', '7894561230', 'farmer2@example.com', 'Farmer Address 2', 'farmerpass2', 'farmer'),

('Company2', '4455667788', 'company2@example.com', 'Company Address 2', 'companypass2', 'company'),

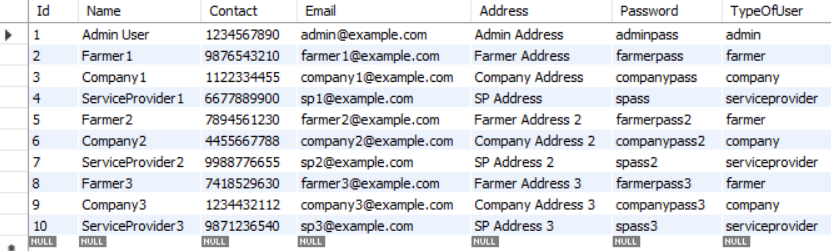
('ServiceProvider2', '9988776655', 'sp2@example.com', 'SP Address 2', 'spass2', 'serviceprovider'),

('Farmer3', '7418529630', 'farmer3@example.com', 'Farmer Address 3', 'farmerpass3', 'farmer'),

('Company3', '1234432112', 'company3@example.com', 'Company Address 3', 'companypass3', 'company'),

('ServiceProvider3', '9871236540', 'sp3@example.com', 'SP Address 3', 'spass3', 'serviceprovider');

**Design:**

****

**2.Insert data in property table:**

INSERT INTO property (FarmerId, Location, TypeOfLand, ImgOfLand, ImgOf7\_12, AreaInAcre, AreaInGuntha) VALUES

(2, 'Village A', 'bagayti', 'land1.jpg', 'doc1.jpg', 5, 10),

(2, 'Village B', 'jirayati', 'land2.jpg', 'doc2.jpg', 3, 5),

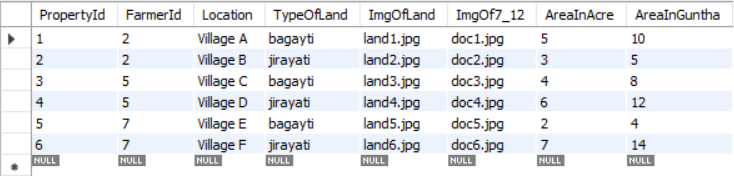
(5, 'Village C', 'bagayti', 'land3.jpg', 'doc3.jpg', 4, 8),

(5, 'Village D', 'jirayati', 'land4.jpg', 'doc4.jpg', 6, 12),

(7, 'Village E', 'bagayti', 'land5.jpg', 'doc5.jpg', 2, 4),

(7, 'Village F', 'jirayati', 'land6.jpg', 'doc6.jpg', 7, 14);

**Design:**



**3.Insert data in application table:**

INSERT INTO application (CompanyId, PropertyId, Status) VALUES

(3, 1, 'pending')

(3, 2, 'accepted'),

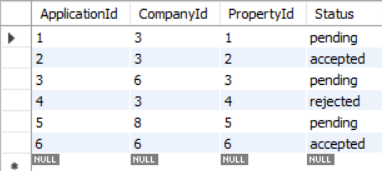
(6, 3, 'pending'),

(3, 4, 'rejected'),

(8, 5, 'pending'),

(6, 6, 'accepted');

**Design:**



**4.Insert data in service table:**

INSERT INTO service (ServiceProviderId, Name, ServiceName, Price, Description) VALUES

(4, 'John Doe', 'Tractor Rental', 500.00, 'Rent a tractor for farming')

(4, 'John Doe', 'Plowing Service', 700.00, 'Plowing for better soil preparation'),

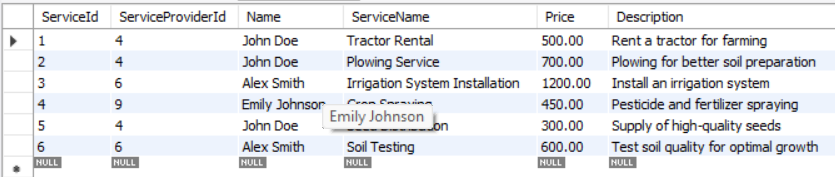
(6, 'Alex Smith', 'Irrigation System Installation', 1200.00, 'Install an irrigation system'),

(9, 'Emily Johnson', 'Crop Spraying', 450.00, 'Pesticide and fertilizer spraying'),

(4, 'John Doe', 'Seed Distribution', 300.00, 'Supply of high-quality seeds'),

(6, 'Alex Smith', 'Soil Testing', 600.00, 'Test soil quality for optimal growth');

**Design:**



**5.Insert data in service\_application table:**

INSERT INTO service\_applications (ServiceId, CompanyId, Status, IsComplete) VALUES

(1, 3, 'pending', FALSE)

(2, 3, 'accepted', TRUE),

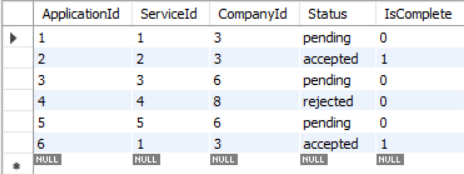
(3, 6, 'pending', FALSE),

(4, 8, 'rejected', FALSE),

(5, 6, 'pending', FALSE),

(1, 3, 'accepted', TRUE);

**Design:**



**6.Insert data in contact table:**

INSERT INTO contact (Name, Email, Description) VALUES

('User1', 'user1@example.com', 'Inquiry about services')

('User2', 'user2@example.com', 'Query regarding land registration'),

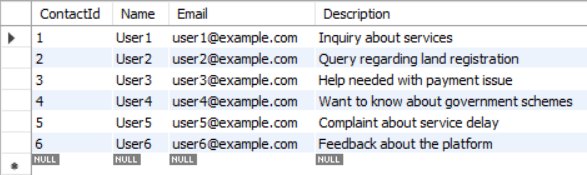
('User3', 'user3@example.com', 'Help needed with payment issue'),

('User4', 'user4@example.com', 'Want to know about government schemes'),

('User5', 'user5@example.com', 'Complaint about service delay'),

('User6', 'user6@example.com', 'Feedback about the platform');

**Design:**



**7.Insert data in payment table:**

INSERT INTO payment (ServiceOrPropertyId, PaymentMethod, PaymentMode, PaymentStatus) VALUES

(1, 'UPI', 'Online', 'paid')

(2, 'Credit Card', 'Online', 'paid'),

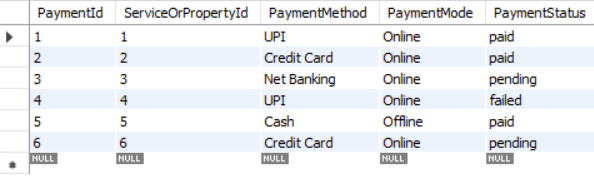
(3, 'Net Banking', 'Online', 'pending'),

(4, 'UPI', 'Online', 'failed'),

(5, 'Cash', 'Offline', 'paid'),

(6, 'Credit Card', 'Online', 'pending');

**Design:**



**8.Insert data in government Schemas table:**

INSERT INTO governmentSchemes (Title, Price, StartDate, LastDate, Description) VALUES

('Organic Farming Subsidy', 20000.00, '2025-01-01', '2025-12-31', 'Subsidy for organic farming.')

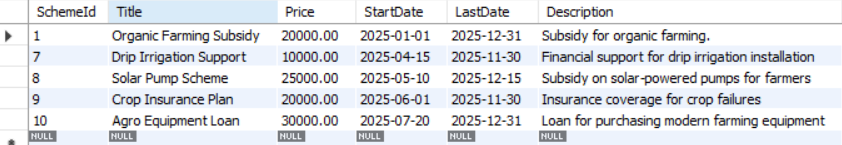
('Drip Irrigation Support', 10000.00, '2025-04-15', '2025-11-30', 'Financial support for drip irrigation installation'),

'Solar Pump Scheme', 25000.00, '2025-05-10', '2025-12-15', 'Subsidy on solar-powered pumps for farmers'),

('Crop Insurance Plan', 20000.00, '2025-06-01', '2025-11-30', 'Insurance coverage for crop failures'),

('Agro Equipment Loan', 30000.00, '2025-07-20', '2025-12-31', 'Loan for purchasing modern farming equipment');

**Design:**

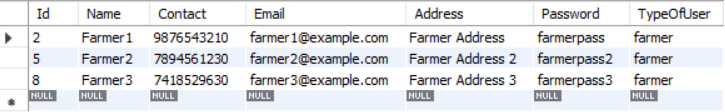


**Fetch the data from the database**

**1. Fetch all farmers**

SELECT \* FROM user WHERE TypeOfUser = 'farmer';

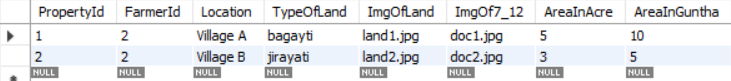
**Design:**



**2. Fetch properties added by a specific farmer**

SELECT \* FROM property WHERE FarmerId = (SELECT Id FROM user WHERE Name = 'Farmer1');

**Design:**



**3. Fetch applications submitted by companies for properties**

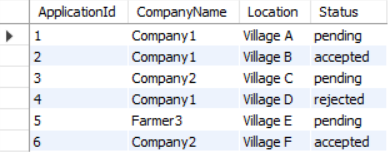
SELECT a.ApplicationId, u.Name AS CompanyName, p.Location, a.Status

FROM application a

JOIN user u ON a.CompanyId = u.Id

JOIN property p ON a.PropertyId = p.PropertyId;

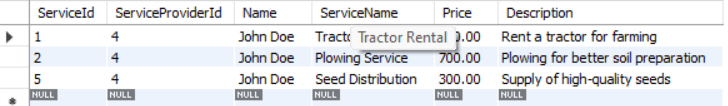
**Design:**



**4. Fetch services offered by a service provider**

SELECT \* FROM service WHERE ServiceProviderId = (SELECT Id FROM user WHERE Name = 'ServiceProvider1');

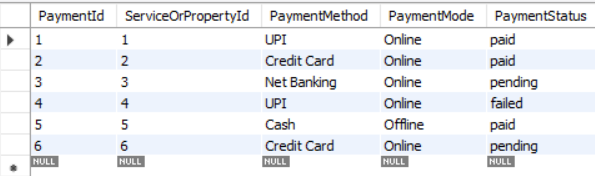
**Design:**



**5. Fetch payments made for property registration**

SELECT \* FROM payment WHERE ServiceOrPropertyId IN (SELECT PropertyId FROM property);

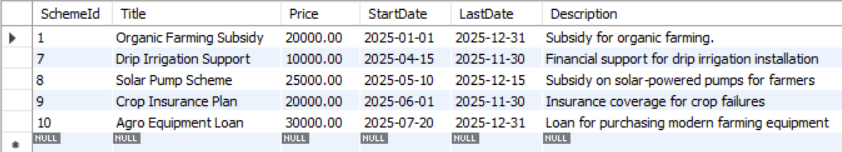
**Design:**



**6. Fetch government schemes available**

SELECT \* FROM governmentSchemes WHERE LastDate >= CURDATE();

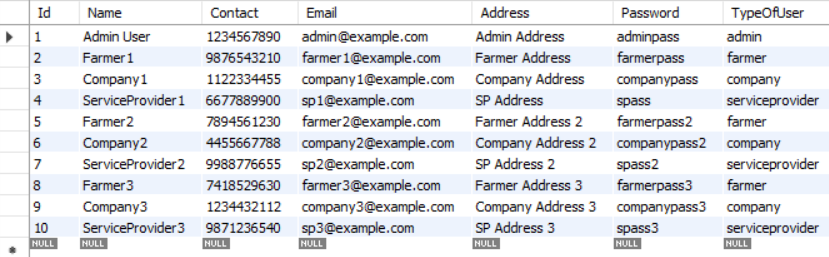
**Design:**

****

**7. Fetch all users data**

SELECT \* FROM user;

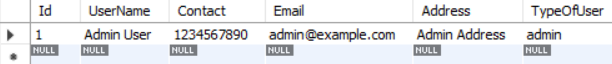
Design:

****

**8. fetch user data and show in the profile:**

SELECT Id, Name AS UserName, Contact, Email, Address, TypeOfUser FROM user WHERE Id = ?;

**Design:**

****

**9. fetch data for the**