

Page 10 of 10



- 1

- The *Household* entity aggregates citizens and their financial details and participates in *Census* data..
- Each *Asset* is owned by either a *Citizen* or the *Panchayat*, which is indicated by the descriptive attribute *ownership\_type*.
- *Land* is a specialization of *Asset*, which is linked to *Crop* and participates in *Agricultural & Environmental* data.
- *Citizens' Income* and *Assets* generate *Tax* which is levied by the *Panchayat*.
- *Citizens* can hold multiple *Certificates* and subscribe to *Welfare Schemes*, both provided by the *Panchayat*.

#### Normalization:

Proper entity separation for scalability (citizens, households, land, etc.).

Avoids redundancy and maintains data integrity.

#### Modular Structure:

Divided based on major domains (citizen-centric, agricultural, administrative, financial).

Easily extendable for additional attributes or entities.

### ER Diagram → Relations (Tables)

- Each entity is converted into a relation
- Multivalued attributes are represented in separate relations
- Every relationship is represented either by a separate relation or by a Foreign-Key constraint
- Composite attributes are broken down into constituent attributes

Entities: *Citizen*, *Household*; Relationships: *belongs*

```
CREATE TABLE citizens (
    citizen_id SERIAL PRIMARY KEY,
    first_name VARCHAR(50), -- Composite attribute 'name' broken down into
    'first_name', 'middle_name', 'last_name'
    middle_name VARCHAR(50),
    last_name VARCHAR(50),
    gender VARCHAR(6),
    dob DATE,
    educational_qualification VARCHAR(50),
    household_id INT REFERENCES households(household_id) -- Citizen <belongs> Household
);

CREATE TABLE phone_numbers ( -- Multivalued attribute 'phone_number' of Citizen
    phone_id SERIAL PRIMARY KEY,
    citizen_id INT REFERENCES citizens(citizen_id),
    phone_number VARCHAR(15)
);

CREATE TABLE households (
    household_id SERIAL PRIMARY KEY,
    address_id INT REFERENCES addresses(address_id) -- Composite attribute Address
```

```
);
CREATE TABLE addresses{
  address_id SERIAL PRIMARY KEY,
  house_number VARCHAR(10),
  street_name VARCHAR(50),
  village VARCHAR(50),
  district VARCHAR(50),
  state VARCHAR(50),
}
```

Entities: *Panchayat*, *Panchayat Member*; Relationships: *works*

```
CREATE TABLE panchayat{
  panchayat_id SERIAL PRIMARY KEY,
  address_id INT REFERENCES addresses(address_id) -- Composite attribute Address
  jurisdiction_area VARCHAR(50),
}
CREATE TABLE panchayat_members (
  employee_id SERIAL PRIMARY KEY,
  citizen_id INT REFERENCES citizens(citizen_id), -- Panchayat_Member <is> Citizen
  role VARCHAR(50)
  joining_date DATE
  panchayat_id INT REFERENCES panchayat(panchayat_id) -- Panchayat_Member <works>
  Panchayat
);
```

Entities: *Welfare Scheme*; Relationships: *subscribes*, *provides*

```
CREATE TABLE welfare_schemes (
  scheme_id SERIAL PRIMARY KEY,
  name VARCHAR(50),
  description TEXT
  panchayat_id INT REFERENCES panchayat(panchayat_id) -- Panchayat <provides>
  Welfare_Scheme
);
CREATE TABLE scheme_subscriptions ( -- Citizen <subscribes> Welfare_Scheme
  subscription_id SERIAL PRIMARY KEY,
  citizen_id INT REFERENCES citizens(citizen_id),
  scheme_id INT REFERENCES welfare_schemes(scheme_id),
  start_date DATE,
  end_date DATE
);
```

Entities: *Certificate*; Relationships: *holds*, *issues*

```

CREATE TABLE certificates (
  certificate_id SERIAL PRIMARY KEY,
  type VARCHAR(50),
  issue_date DATE,
  expiry_date DATE,
  citizen_id INT REFERENCES citizens(citizen_id) -- Citizen <holds> Certificate
  panchayat_id INT REFERENCES panchayat(panchayat_id) -- Panchayat <issues>
Certificate
);

```

Entities: Asset; Relationships: *owns*, *generates*

```

CREATE TABLE assets{
  asset_id SERIAL PRIMARY KEY,
  tax_id INT REFERENCES tax_records(tax_id), -- Asset <generates> Tax
  type VARCHAR(50),
}
CREATE TABLE asset_ownership ( -- Panchayat or Citizen <owns> Asset
  ownership_id SERIAL PRIMARY KEY,
  asset_id INT REFERENCES assets(asset_id),
  ownership_type VARCHAR(50) NOT NULL CHECK (ownership_type IN ('Private', 'Govt')),
  citizen_id INT REFERENCES citizens(citizen_id),
  panchayat_id INT REFERENCES panchayat(panchayat_id),
  CHECK (
    (ownership_type = 'Private' AND citizen_id IS NOT NULL AND panchayat_id IS
NULL)
    OR
    (ownership_type = 'Govt' AND panchayat_id IS NOT NULL AND citizen_id IS NULL)
  )
);

```

Entities: *Land*, *Crop*; Relationships: *grown*

```

CREATE TABLE land_records{
  land_id SERIAL PRIMARY KEY,
  location_id INT REFERENCES addresses(address_id) -- Composite attribute Location
  size NUMERIC(10, 2),
  land_type VARCHAR(50),
  soil_type VARCHAR(50),
  asset_id INT REFERENCES assets(asset_id) -- Land <is> Asset
  crop_id INT REFERENCES crops(crop_id) -- Land <grows> Crop
}
CREATE TABLE crops{
  crop_id SERIAL PRIMARY KEY,
  name VARCHAR(50),

```

```

    type VARCHAR(50),
    season VARCHAR(50),
    yield NUMERIC(10, 2)
    planting_date DATE,
    harvest_date DATE
}

```

Entities: *Income, Tax, Expenditure*; Relationships: *earns, generates, levies, performs*

```

CREATE TABLE income{
    income_id SERIAL PRIMARY KEY,
    citizen_id INT REFERENCES citizens(citizen_id), -- Citizen <earns> Income
    source VARCHAR(50),
    amount NUMERIC(10, 2)
    tax_id INT REFERENCES tax_records(tax_id) -- Income <generates> Tax
}
CREATE TABLE tax_records {
    tax_id SERIAL PRIMARY KEY,
    tax_type VARCHAR(50),
    rate NUMERIC(10, 2),
    panchayat_id INT REFERENCES panchayat(panchayat_id) -- Panchayat <levies> Tax
}
CREATE TABLE expenditure{
    expenditure_id SERIAL PRIMARY KEY,
    type VARCHAR(50),
    amount NUMERIC(10, 2),
    date DATE
    citizen_id INT REFERENCES citizens(citizen_id) -- Citizen <performs> Expenditure
}

```

Entities: *Data, Census, Agricultural, Environmental*; Relationships: *participates, surveys*

```

CREATE TABLE data_details{
    data_id SERIAL PRIMARY KEY,
    department VARCHAR(50),
    area VARCHAR(50),
    survey_date DATE,
}
CREATE TABLE census_data{ -- Citizen or Household <participates> Census
    data_id INT REFERENCES data_details(data_id), -- Census_Data <is> Data
    type VARCHAR(50),
    demographic VARCHAR(50),
    count INT,
    level VARCHAR(50) NOT NULL CHECK (level IN ('Household', 'Citizen')),
    citizen_id INT REFERENCES citizens(citizen_id),
}

```

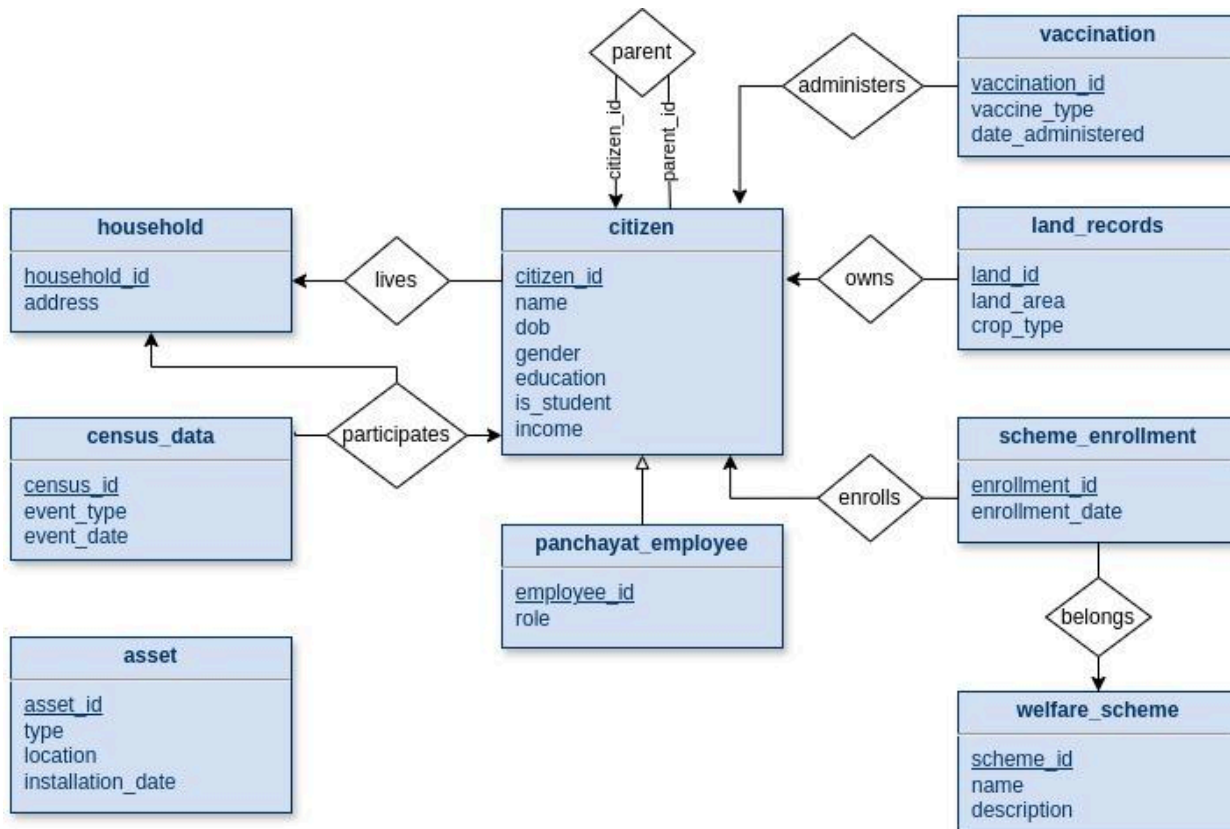
```

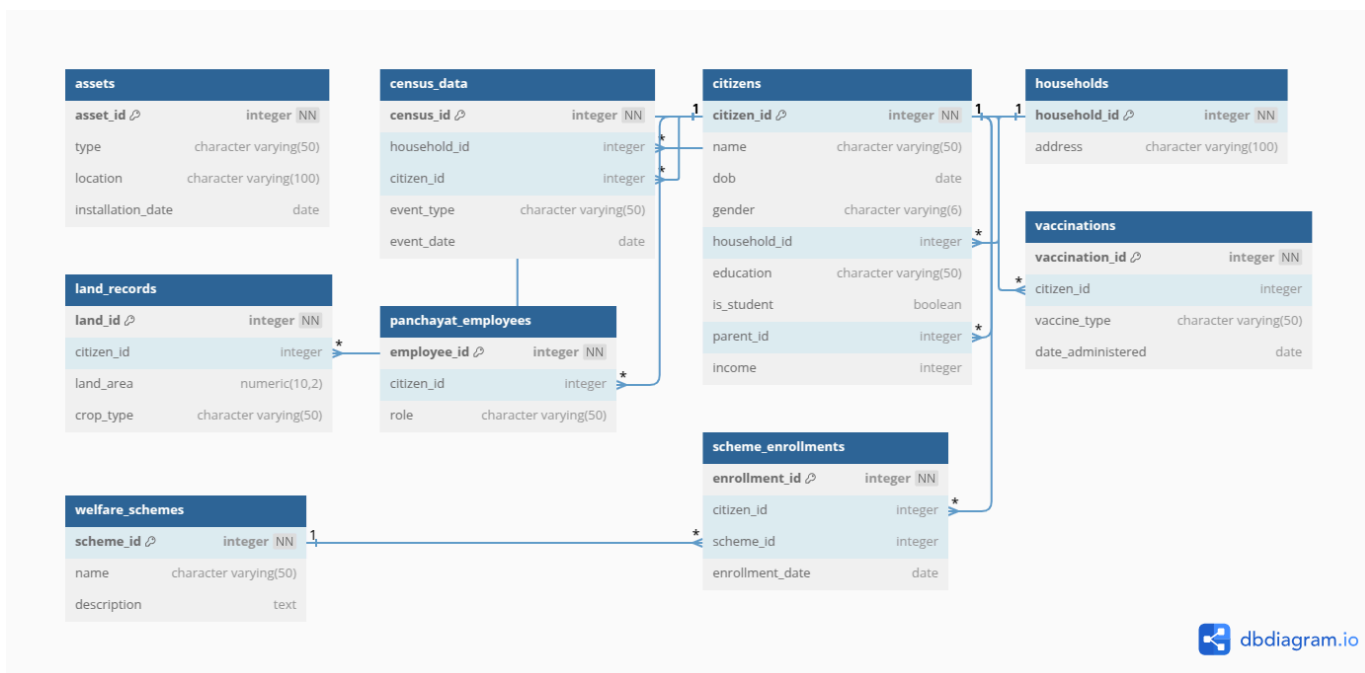
household_id INT REFERENCES households(household_id),
CHECK (
    (level = 'Household' AND citizen_id IS NULL AND household_id IS NOT NULL) OR
    (level = 'Citizen' AND citizen_id IS NOT NULL AND household_id IS NULL)
)
panchayat_id INT REFERENCES panchayat(panchayat_id) -- Panchayat <surveys>
Census_Data
}
CREATE TABLE agricultural_data{
    data_id INT REFERENCES data_details(data_id), -- Agricultural_Data <is> Data
    crop_id INT REFERENCES crops(crop_id), -- Crop <participates> Agricultural_Data
    land_id INT REFERENCES land_records(land_id), -- Land <participates>
Agricultural_Data
    irrigation_details VARCHAR(50),
    fertilizer_usage VARCHAR(50),
    pesticide_usage VARCHAR(50),
    machinery_usage VARCHAR(50),
}
CREATE TABLE environmental_data{
    data_id INT REFERENCES data_details(data_id), -- Environmental_Data <is> Data
    flora VARCHAR(50), -- Composite attribute 'biodiversity' broken down into 'flora',
'fauna'
    fauna VARCHAR(50),
    water_table NUMERIC(10, 2),
    land_use VARCHAR(50),
    land_id INT REFERENCES land_records(land_id), -- Land <participates>
Environmental_Data
}

```

## Schemas Used for this assignment

### ER Diagram





## SQL Queries to Create Tables

```

CREATE TABLE households (
    household_id SERIAL PRIMARY KEY,
    address VARCHAR(100)
);

CREATE TABLE citizens (
    citizen_id SERIAL PRIMARY KEY,
    name VARCHAR(50),
    dob DATE,
    gender VARCHAR(6),
    household_id INT REFERENCES households(household_id),
    education VARCHAR(50),
    is_student BOOLEAN,
    parent_id INT REFERENCES citizens(citizen_id),
    income INT
);

CREATE TABLE land_records (
    land_id SERIAL PRIMARY KEY,
    citizen_id INT REFERENCES citizens(citizen_id),
    land_area NUMERIC(10, 2),
    crop_type VARCHAR(50)
);

CREATE TABLE panchayat_employees (
    employee_id SERIAL PRIMARY KEY,
    citizen_id INT REFERENCES citizens(citizen_id),

```



```
    role VARCHAR(50)
);
CREATE TABLE assets (
    asset_id SERIAL PRIMARY KEY,
    type VARCHAR(50),
    location VARCHAR(100),
    installation_date DATE
);
CREATE TABLE welfare_schemes (
    scheme_id SERIAL PRIMARY KEY,
    name VARCHAR(50),
    description TEXT
);
CREATE TABLE scheme_enrollments (
    enrollment_id SERIAL PRIMARY KEY,
    citizen_id INT REFERENCES citizens(citizen_id),
    scheme_id INT REFERENCES welfare_schemes(scheme_id),
    enrollment_date DATE
);
CREATE TABLE vaccinations (
    vaccination_id SERIAL PRIMARY KEY,
    citizen_id INT REFERENCES citizens(citizen_id),
    vaccine_type VARCHAR(50),
    date_administered DATE
);
CREATE TABLE census_data (
    census_id SERIAL PRIMARY KEY,
    household_id INT REFERENCES households(household_id),
    citizen_id INT REFERENCES citizens(citizen_id),
    event_type VARCHAR(50),
    event_date DATE
);
```

## SQL Queries

A. Show names of all citizens who hold more than 1 acre of land

```
SELECT c.name
FROM citizens AS c
JOIN land_records AS l ON c.citizen_id = l.citizen_id
WHERE l.land_area > 1.00;
```

B. Show name of all girls who study in school with household income less than 1 Lakh per year

```
SELECT c1.name
FROM citizens AS c1
JOIN households AS h ON c1.household_id = h.household_id
WHERE c1.gender = 'Female'
AND c1.is_student = TRUE
AND
( SELECT SUM(c2.income)
FROM citizens AS c2
WHERE c2.household_id = h.household_id
)
< 100000;
```

C. How many acres of land cultivate rice

```
SELECT SUM(land_area)
FROM land_records
WHERE crop_type ILIKE 'rice';
```

D. Number of citizens who are born after 1.1.2000 and have educational qualification of 10th class

```
SELECT COUNT(*)
FROM citizens
WHERE dob > '2000-01-01'
AND education ILIKE '10th';
```

E. Name of all employees of panchayat who also hold more than 1 acre land

```
SELECT c.name
FROM citizens AS c
JOIN panchayat_employees AS p ON c.citizen_id = p.citizen_id
JOIN land_records AS l ON p.citizen_id = l.citizen_id
WHERE l.land_area > 1.00;
```

F. Name of the household members of Panchayat Pradhan

```
SELECT c1.name
FROM citizens AS c1
JOIN households AS h ON h.household_id = c1.household_id
JOIN citizens AS c2 ON c2.household_id = h.household_id
JOIN panchayat_employees AS p ON p.citizen_id = c2.citizen_id
WHERE p.role ILIKE 'Pradhan';
```

G. Total number of street light assets installed in a particular locality named Phulera that are installed in 2024

```
SELECT COUNT(*)
FROM assets
WHERE type ILIKE 'Street Light'
AND location ILIKE 'Phulera'
AND EXTRACT(YEAR FROM installation_date) = 2024;
```

H. Number of vaccinations done in 2024 for the children of citizens whose educational qualification is class 10

```
SELECT COUNT(*)
FROM vaccinations AS v
JOIN citizens AS c1 ON v.citizen_id = c1.citizen_id
JOIN citizens AS c2 ON c1.parent_id = c2.citizen_id
WHERE EXTRACT(YEAR FROM v.date_administered) = 2024
AND c2.education ILIKE '10th';
```

I. Total number of births of boy child in the year 2024

```
SELECT COUNT(*) FROM citizens
WHERE gender = 'Male'
AND EXTRACT(YEAR FROM dob) = 2024;
```

J. Number of citizens who belong to the household of at least one panchayat employee.

```
SELECT COUNT(DISTINCT c1)
FROM citizens AS c1
JOIN households AS h ON h.household_id = c1.household_id
JOIN citizens AS c2 ON c2.household_id = h.household_id
JOIN panchayat_employees AS p ON p.citizen_id = c2.citizen_id;
```