DBMS Lab Assignment 2: Gram Panchayat Management System

Dadi Sasank Kumar
Roll Number: 22CS10020

January 28, 2025

Entity-Relationship Diagram and Table Schema

Entity Sets

Citizen	(Aadhar_no, first_name, last_name, date_of_birth, phone_number, age, caste)
Census_data	(census_id, year, population, demographics, description)
Environmental_data	(report_env_id, issue_type, description, report_date, rainfall, groundwater_level, pollution_data)
Asset	(asset_id, type, value, location, name)
Expenditure	(expenditure_id, amount, expenditure_date, purpose, category, payment_mode, financial_year)
Income	(income_id, amount, income_date, source, financial_year)
Tax	(tax_id, amount, payment_date, status, payment_mode, financial_year, citizen_id)
Service	(service_id, type, status, request_date)
Welfare_Scheme	(scheme_id, name, description, start_date, end_date)
Agricultural_data	(report_agri_id, crop_type, area, yield, report_date, citizen_id)
Panchayat_committee	(committee_id, name, purpose)
Panchayat_member	(member_id, first_name, last_name, role, contact)

Relationship Sets

Reports	(citizen_id, report_id)
Benefits	(scheme_id, citizen_id)
Requests	(service_id, citizen_id)
Pays	(tax_id, citizen_id)
Earns	(income_id, citizen_id)
Incurs	(expenditure_id, citizen_id)
Owns	(asset_id, citizen_id)
Submits	(citizen_id, report_id)
Includes	(census_id, citizen_id)
Belongs	(committee_id, member_id)
Oversees	(committee_id, . . . , census_id)

Aggregation

Oversees: The "Oversees" relationship aggregates multiple activities and entities under the supervision of the Panchayat Committee. This relationship consolidates various aspects such as agricultural reports, environmental reports, taxes, services, welfare schemes, and census data of citizens. Aggregating these relationships simplifies the representation of complex interdependencies, ensuring efficient management and accountability by the Panchayat Committee.

Appropriateness of Schemas for Gram Panchayat Management System

The schemas and relationships defined above are well-suited for a Gram Panchayat Management System because they comprehensively capture the essential entities, attributes, and interconnections relevant to rural governance and administration. Below is an explanation of why these schemas are appropriate:

Entity Sets

- **Citizen:** This entity serves as the core of the system, encapsulating personal and demographic details of individuals residing within the Panchayat jurisdiction. Attributes such as *Aadhar_no* ensure unique identification, while others like *age*, *caste*, and *phone_number* allow for targeted welfare distribution, demographic analysis, and efficient communication.
- **Census_data:** This entity tracks population growth, demographic changes, and other statistical insights critical for planning, resource allocation, and development activities.
- **Environmental_data:** This entity records essential environmental metrics such as *rainfall*, *groundwater_level*, and *pollution_data*, enabling data-driven decisions for sustainable environmental management and agricultural planning.
- **Asset:** Tracks movable and immovable assets owned by the Gram Panchayat and its residents. Attributes like *type*, *value*, and *location* help in asset management and governance.
- **Expenditure:** Captures the details of financial outflows, helping the Panchayat track spending on public infrastructure, welfare schemes, and administrative operations.
- **Income:** Records financial inflows such as taxes, grants, and other sources of revenue, ensuring financial transparency and accountability.
- **Tax:** Stores detailed records of taxes paid by citizens, facilitating efficient collection and management of Panchayat funds.
- **Service:** Tracks citizen service requests, such as infrastructure maintenance, grievance redressal, and utility services, ensuring effective service delivery.
- **Welfare_Scheme:** Documents government-initiated welfare schemes, including details such as *name*, *description*, and *start_date*, ensuring that citizens benefit from these programs.
- **Agricultural_data:** Records agricultural metrics, including *crop_type*, *area*, and *yield*, which are essential for planning irrigation, subsidies, and crop insurance programs.
- **Panchayat_committee:** Represents the governing body responsible for decision-making and oversight of all Gram Panchayat operations.
- **Panchayat_member:** Captures the details of committee members, ensuring transparency in the roles and responsibilities of individuals in governance.

Relationship Sets

- **Reports:** Connects citizens with various reports they submit, such as environmental or agricultural data, ensuring traceability and accountability.
- **Benefits:** Links welfare schemes with citizens who benefit from them, facilitating effective distribution and tracking of resources.
- **Requests:** Tracks service requests made by citizens, ensuring efficient service delivery and response management.
- **Pays:** Connects tax payments to citizens, ensuring accountability in tax collection.
- **Earns:** Links citizen income with its sources, enabling better financial planning and monitoring.
- **Incurs:** Tracks expenditures incurred by citizens, facilitating financial transparency.
- **Owns:** Links assets to their respective owners, ensuring clarity in asset ownership.
- **Submits:** Tracks report submissions made by citizens, linking them to environmental or agricultural data.
- **Includes:** Associates census data with citizens, allowing for demographic analysis and planning.

- **Belongs:** Links Panchayat committee members to their respective committees, ensuring clarity in roles and responsibilities.
- **Oversees:** Aggregates multiple activities and entities under the supervision of the Panchayat Committee. This relationship simplifies the representation of complex interdependencies such as agricultural reports, environmental reports, taxes, services, welfare schemes, and census data, ensuring efficient management and accountability.

Relationship Sets and Cardinalities

Reports	One-to-Many (1:N): Each citizen (1..1) can submit multiple (0..*) agricultural reports
Benefits	One-to-Many (1:N): Each citizen (1..1) can benefit from multiple (0..*) welfare schemes
Requests	One-to-Many (1:N): Each citizen (1..1) can request multiple (0..*) services
Pays	One-to-Many (1:N): Each citizen (1..1) can have multiple (0..*) tax records
Earns	One-to-Many (1:N): Each citizen (1..1) can have multiple (0..*) income records
Incurs	One-to-Many (1:N): Each citizen (1..1) can have multiple (0..*) expenditure records
Owns	One-to-Many (1:N): Each citizen (1..1) can own multiple (0..*) assets
Submits	One-to-Many (1:N): Each citizen (1..1) can submit multiple (0..*) environmental reports
Includes	One-to-Many (1:N): Each citizen (1..1) must be included in multiple (0..*) census records
Belongs	One-to-Many (1:N): Each Panchayat member (1..1) belongs to multiple (0..*) committees
Oversees	Many-to-One (N:1): The Panchayat committee (1..*) oversees one (1..1) aggregation of citizen-related entities

Justification for Cardinality Relationships

- **One-to-Many (1..1 to 0..*) Relationships:**
  - **Reports/Agricultural Data:** Each citizen is uniquely identified and can submit multiple agricultural reports over time, but each report must be associated with exactly one citizen.
  - **Benefits/Welfare Scheme:** A citizen can be eligible for zero or more welfare schemes, while maintaining their unique identity in the system.
  - **Requests/Service:** A citizen may request multiple services or none at all, while maintaining a single identity in the system.
  - **Financial Relationships (Pays, Earns, Incurs):** Each citizen maintains a single identity while potentially having multiple financial records (taxes, income, expenditure).
  - **Owns/Asset:** A citizen can own multiple assets or none, while maintaining a single identity record.
  - **Submits/Environmental Data:** Each citizen can submit multiple environmental reports over time, with each submission linked to their unique identity.
  - **Includes/Census:** A citizen must be uniquely identified in the system but may appear in multiple census records over different years.
  - **Belongs:** Each Panchayat member has a unique identity and can belong to multiple committees.
- **Many-to-One (N:1) for Oversight:**
  - **Oversees:** Multiple Panchayat committee members (1..*) collectively oversee one unified aggregation (1..1) of citizen-related entities and activities. This aggregation includes:
    - Agricultural reports
    - Environmental reports
    - Tax records
    - Service requests
    - Welfare scheme distributions
    - Census data

Implementation Considerations

The cardinality constraints inform these key implementation decisions:

- All one-to-many relationships (1..1 to 0..*) should be implemented using foreign keys in the "many" side tables, referencing the primary key of the "one" side
- The citizen's unique identifier (Aadhar_no) serves as the primary key and will be referenced by multiple related tables
- The Oversees aggregation relationship requires careful implementation to maintain the integrity of the collective oversight while allowing multiple committee members to participate
- Nullable foreign keys should be used where the 0..* cardinality indicates optional relationships
- Referential integrity constraints must be implemented to ensure:
  - No orphaned records can exist in the "many" side tables
  - Updates to citizen records are properly cascaded where appropriate
  - Deletion rules protect critical historical data while maintaining data consistency

The defined schemas and relationships holistically represent the key aspects of governance, administration, and citizen welfare in a Gram Panchayat Management System.

Table Schema Explanation

Households Table

```
CREATE TABLE households(
    household_id INTEGER PRIMARY KEY,
    address VARCHAR(200) NOT NULL,
    number_of_members INTEGER NOT NULL,
    income INTEGER
);
```

- **Households Table**: Stores details about households such as address, number of members, and income.

Citizens Table

```
CREATE TABLE citizens(
    citizen_id INTEGER PRIMARY KEY,
    aadhar_no VARCHAR(200) UNIQUE NOT NULL,
    first_name VARCHAR(200) NOT NULL,
    last_name VARCHAR(200) NOT NULL,
    date_of_birth DATE NOT NULL,
    phone_number VARCHAR(200),
    age INTEGER,
    caste VARCHAR(200),
    gender VARCHAR(200) NOT NULL,
    household_id INTEGER NOT NULL,
    FOREIGN KEY (household_id) REFERENCES households(household_id),
    educational_qualification VARCHAR(200),
    occupation VARCHAR(200),
    marital_status VARCHAR(200)
);
```

- **Citizens Table**: Stores information about citizens, including personal details, household references, and more.

Land Records Table

```
CREATE TABLE land_records(
    land_id INTEGER PRIMARY KEY,
    citizen_id INTEGER,
    FOREIGN KEY (citizen_id) REFERENCES citizens(citizen_id),
    area_acres DECIMAL NOT NULL,
    crop_type VARCHAR(200)
);
```

- **Land Records Table**: Tracks land owned by citizens, including area in acres and crop type.

Panchayat Employees Table

```

create table panchayat_employees(
    employee_id INTEGER PRIMARY KEY,
    citizen_id INTEGER,
    FOREIGN KEY (citizen_id) REFERENCES citizens(citizen_id),
    role VARCHAR(200)
);

ALTER TABLE panchayat_employees
ADD COLUMN household_id INTEGER;

UPDATE panchayat_employees pe
SET household_id = h.household_id
FROM households h
WHERE pe.employee_id = h.household_id;
);
```

- **Panchayat Employees Table**: Stores information about employees working for the panchayat, linked to citizens.

Assets Table

```
CREATE TABLE assets(
    asset_id INTEGER PRIMARY KEY,
    type VARCHAR(200) NOT NULL,
    household_id INTEGER,
    FOREIGN KEY (household_id) REFERENCES households(household_id),
    value DECIMAL NOT NULL,
    location VARCHAR(200),
    name VARCHAR(200),
    installation_date DATE
);
```

- **Assets Table**: Records assets owned by households, including type, value, and installation date.

Welfare Schemes Table

```
CREATE TABLE welfare_schemes(
    scheme_id INTEGER PRIMARY KEY,
    name VARCHAR(200) NOT NULL,
    description VARCHAR(200),
    eligibility VARCHAR(200),
    benefits VARCHAR(200),
    start_date DATE,
    end_date DATE
);
```

- **Welfare Schemes Table**: Stores details about welfare schemes, including eligibility and benefits.

Scheme Enrollments Table

```
CREATE TABLE scheme_enrollments(
    enrollment_id INTEGER PRIMARY KEY,
    citizen_id INTEGER,
    FOREIGN KEY (citizen_id) REFERENCES citizens(citizen_id),
    scheme_id INTEGER,
    FOREIGN KEY (scheme_id) REFERENCES welfare_schemes(scheme_id),
    enrollment_date DATE NOT NULL
);
```

- **Scheme Enrollments Table**: Tracks which citizens are enrolled in which welfare schemes.

Vaccinations Table

```
CREATE TABLE vaccinations(
    vaccination_id INTEGER PRIMARY KEY,
    citizen_id INTEGER,
    FOREIGN KEY (citizen_id) REFERENCES citizens(citizen_id),
    vaccine_type VARCHAR(200) NOT NULL,
    date DATE NOT NULL
);
```

- **Vaccinations Table**: Tracks vaccinations received by citizens, including the vaccine type and date.

Census Data Table

```
CREATE TABLE census_data(
    census_id INTEGER PRIMARY KEY,
    household_id INTEGER,
    FOREIGN KEY (household_id) REFERENCES households(household_id),
    citizen_id INTEGER,
    FOREIGN KEY (citizen_id) REFERENCES citizens(citizen_id),
    year INTEGER NOT NULL,
    population INTEGER NOT NULL,
    demographics VARCHAR(200),
    description VARCHAR(200),
    event_type VARCHAR(200),
    event_date DATE
);
```

- **Census Data Table**: Stores census information, including population and demographic details for each year.

Tax Table

```
CREATE TABLE tax(
    tax_id INTEGER PRIMARY KEY,
    amount DECIMAL NOT NULL,
    payment_date DATE NOT NULL,
    status VARCHAR(200),
    payment_mode VARCHAR(200),
    financial_year VARCHAR(200),
    citizen_id INTEGER,
    FOREIGN KEY (citizen_id) REFERENCES citizens(citizen_id)
);
```

- **Tax Table**: Tracks tax payments made by citizens, including amount, status, and financial year.

Income Table

```
CREATE TABLE income(
    income_id INTEGER PRIMARY KEY,
    citizen_id INTEGER,
    FOREIGN KEY (citizen_id) REFERENCES citizens(citizen_id),
    amount DECIMAL NOT NULL,
    income_date DATE NOT NULL,
    source VARCHAR(200),
    financial_year VARCHAR(200)
);
```

- **Income Table**: Records income details of citizens, including amount, source, and financial year.

Expenditure Table

```
CREATE TABLE expenditure(
    expenditure_id INTEGER PRIMARY KEY,
    citizen_id INTEGER,
    FOREIGN KEY (citizen_id) REFERENCES citizens(citizen_id),
    amount DECIMAL NOT NULL,
    expenditure_date DATE NOT NULL,
    purpose VARCHAR(200),
    category VARCHAR(200),
    payment_mode VARCHAR(200),
    financial_year VARCHAR(200)
);
```

- **Expenditure Table**: Tracks citizens' expenditures, including amount, purpose, and financial year.

Service Table

```
CREATE TABLE service(
    service_id INTEGER PRIMARY KEY,
    name VARCHAR(200) NOT NULL,
    description VARCHAR(200),
    type VARCHAR(200) NOT NULL,
    status VARCHAR(200),
    request_date DATE
);
```

- **Service Table**: Stores information about various services available, including type, description, and request date.

Service Requests Table

```
CREATE TABLE service_requests(
    request_id INTEGER PRIMARY KEY,
    citizen_id INTEGER,
    FOREIGN KEY (citizen_id) REFERENCES citizens(citizen_id),
    service_id INTEGER,
    FOREIGN KEY (service_id) REFERENCES service(service_id),
    request_date DATE NOT NULL,
    status VARCHAR(200)
);
```

- **Service Requests Table**: Tracks requests made by citizens for services, including request status.

Data Population Queries

Households Data Insertion

```
INSERT INTO households (household_id, address, number_of_members, income) VALUES
(1, '123 Main Street, Phulera Village, District Rajasthan', 4, 75000),
(2, '45 Gandhi Road, Phulera Village, District Rajasthan', 3, 50000),
(3, '78 Nehru Colony, Phulera Village, District Rajasthan', 5, 100000),
(4, '22 Shastri Nagar, Phulera Village, District Rajasthan', 2, 60000),
(5, '56 Tilak Street, Phulera Village, District Rajasthan', 6, 85000),
(6, '11 Subhash Chowk, Phulera Village, District Rajasthan', 3, 45000),
(7, '33 Azad Lane, Phulera Village, District Rajasthan', 4, 70000),
(8, '89 Patel Marg, Phulera Village, District Rajasthan', 5, 95000);
```

Update Household Data

```
UPDATE households
SET number_of_members = LEAST(number_of_members, 4);
```

Citizens Data Insertion

```
INSERT INTO citizens (
    citizen_id, aadhar_no, first_name, last_name, date_of_birth,
    phone_number, age, caste, gender, household_id,
    educational_qualification, occupation, marital_status
) VALUES
(1, '123456789012', 'Ramesh', 'Kumar', '1980-05-15', '9876543210', 43, 'General', 'Male', 1,
    'Graduate', 'Farmer', 'Married'),
(2, '234567890123', 'Priya', 'Singh', '2005-03-20', '8765432109', 18, 'OBC', 'Female', 1, '
    12th', 'Student', 'Unmarried'),
(3, '345678901234', 'Amit', 'Patel', '1990-11-10', '7654321098', 33, 'SC', 'Male', 2, '10th'
    , 'Worker', 'Married'),
(4, '456789012345', 'Sunita', 'Gupta', '2016-07-25', '6543210987', 9, 'General', 'Female',
    3, 'Graduate', 'Teacher', 'Married'),
(5, '567890123456', 'Rajesh', 'Sharma', '1975-12-05', '5432109876', 48, 'ST', 'Male', 4, '12
    th', 'Driver', 'Married'),
(6, '678901234567', 'Neha', 'Verma', '2024-09-18', '4321098765', 2, 'OBC', 'Female', 5, '
    Graduate', 'Software Engineer', 'Unmarried'),
(7, '789012345678', 'Vikram', 'Singh', '1995-06-30', '3210987654', 28, 'General', 'Male', 6,
    'Diploma', 'Technician', 'Unmarried'),
(8, '890123456789', 'Meera', 'Devi', '1970-02-14', '2109876543', 53, 'SC', 'Female', 7, '10
    th', 'Homemaker', 'Married');
```

Land Records Data Insertion

```
INSERT INTO land_records (land_id, citizen_id, area_acres, crop_type) VALUES
(1, 1, 2.5, 'Rice'),
(2, 3, 1.0, 'Wheat'),
(3, 4, 1.5, 'Sugarcane'),
(4, 5, 0.8, 'Maize'),
(5, 6, 0.5, 'Vegetables');
```

Panchayat Employees Data Insertion

```
INSERT INTO panchayat_employees (employee_id, citizen_id, role) VALUES
(1, 1, 'Sarpanch'),
(2, 4, 'Secretary'),
(3, 3, 'Village Coordinator');

UPDATE panchayat_employees pe
SET household_id = h.household_id
```

```
FROM households h
WHERE pe.employee_id = h.household_id;
```

Assets Data Insertion

```
INSERT INTO assets (asset_id, type, household_id, value, location, name, installation_date)
    VALUES
(1, 'Street Light', 1, 50000, 'Main Road', 'Street Light 1', '2024-01-15'),
(2, 'Water Pump', 2, 75000, 'Community Well', 'Village Water Pump', '2023-11-20'),
(3, 'Community Hall', 3, 500000, 'Village Center', 'Gram Panchayat Hall', '2022-05-10'),
(4, 'Electricity Transformer', 4, 200000, 'Village Outskirts', 'Power Distribution Unit', '
    2023-07-30'),
(5, 'Public Toilet', 5, 100000, 'Market Area', 'Community Sanitation Facility', '2023-09-25'
    );
```

Welfare Schemes Data Insertion

```
INSERT INTO welfare_schemes (scheme_id, name, description, eligibility, benefits, start_date
    , end_date) VALUES
(1, 'Pradhan Mantri Awas Yojana', 'Housing for All', 'Below Poverty Line Families', 'Free
    Housing', '2022-01-01', '2024-12-31'),
(2, 'PM Kisan Samman Nidhi', 'Direct Income Support to Farmers', 'Small and Marginal Farmers
    ', 'Financial Assistance  6000 /Year', '2019-02-24', NULL),
(3, 'National Social Assistance Programme', 'Social Security for Elderly', 'Senior Citizens,
     Widows', 'Pension and Financial Support', '2020-04-01', NULL),
(4, 'Mid-Day Meal Scheme', 'School Nutrition Programme', 'School Children (6-14 years)', '
    Free Nutritious Meal', '2021-06-15', NULL),
(5, 'Pradhan Mantri Matru Vandana Yojana', 'Maternity Benefit Programme', 'Pregnant and
    Lactating Women', 'Cash Incentive  5000 ', '2017-09-01', NULL);
```

Scheme Enrollments Data Insertion

```
INSERT INTO scheme_enrollments (enrollment_id, citizen_id, scheme_id, enrollment_date)
    VALUES
(1, 1, 1, '2022-06-15'),
(2, 3, 2, '2022-07-20'),
(3, 4, 3, '2022-08-10'),
(4, 2, 4, '2022-09-05'),
(5, 5, 5, '2022-10-12');
```

Vaccinations Data Insertion

```
INSERT INTO vaccinations (vaccination_id, citizen_id, vaccine_type, date) VALUES
(1, 2, 'COVID-19', '2024-01-15'),
(2, 3, 'Polio', '2024-02-20'),
(3, 4, 'Hepatitis B', '2023-11-10'),
(4, 5, 'Tetanus', '2023-12-05'),
(5, 6, 'MMR', '2024-03-01');
```

Census Data Insertion

```
INSERT INTO census_data (
   census_id, household_id, citizen_id, year, population,
   demographics, description, event_type, event_date
) VALUES
(1, 1, 1, 2023, 4, 'Rural Family', 'Household Census Data', 'Population Survey', '2023-03-15
    '),
(2, 2, 3, 2023, 3, 'Working Class Family', 'Household Census Data', 'Population Survey', '
    2023-03-16'),
```

```
(3, 3, 4, 2023, 4, 'Middle-Income Family', 'Household Census Data', 'Population Survey', '2023-03-17'),
(4, 4, 5, 2023, 2, 'Small Family', 'Household Census Data', 'Population Survey', '2023-03-18'),
(5, 5, 6, 2023, 3, 'Young Professional Family', 'Household Census Data', 'Population Survey', '2023-03-19');
```

Tax Data Insertion

```
INSERT INTO tax (tax_id, amount, payment_date, status, payment_mode, financial_year, citizen_id)
VALUES
(1, 5000.00, '2023-03-15', 'Paid', 'Online', '2023-2024', 1),
(2, 3000.00, '2023-04-10', 'Pending', 'Cash', '2023-2024', 2),
(3, 7000.00, '2023-05-05', 'Paid', 'Card', '2023-2024', 3),
(4, 4000.00, '2023-06-20', 'Paid', 'UPI', '2023-2024', 4),
(5, 6000.00, '2023-07-25', 'Pending', 'Cheque', '2023-2024', 5);
```

SQL Queries and Corresponding Tasks

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

```
SELECT c.first_name, c.last_name
FROM citizens c
JOIN land_records l ON c.citizen_id = l.citizen_id
WHERE l.area_acres > 1;
```

Task B: Show names of all girls who study in school with household income less than 1 Lakh per year

```
SELECT c.first_name, c.last_name
FROM citizens c
JOIN households h ON c.household_id = h.household_id
WHERE c.gender = 'Female'
  AND c.occupation = 'Student'
  AND h.income < 100000;
```

Task C: How many acres of land cultivate rice

```
SELECT SUM(l.area_acres) AS total_rice_acres
FROM land_records l
WHERE l.crop_type = 'Rice';
```

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

```
SELECT COUNT(*) AS total_citizens
FROM citizens
WHERE date_of_birth > '2000-01-01'
  AND educational_qualification = '10th class';
```

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

```
SELECT c.first_name, c.last_name
FROM citizens c
JOIN panchayat_employees p ON c.citizen_id = p.citizen_id
JOIN land_records l ON c.citizen_id = l.citizen_id
WHERE l.area_acres > 1;
```

Task F: Name of the household members of Panchayat Pradhan

```
SELECT c.first_name, c.last_name
FROM citizens c
WHERE c.household_id = (
    SELECT p.household_id
    FROM panchayat_employees p
    JOIN citizens pc ON p.citizen_id = pc.citizen_id
    WHERE p.role = 'Sarpanch'
);
```

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

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

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

```sql
SELECT COUNT(*) AS total_vaccinations
FROM vaccinations v
JOIN citizens c ON v.citizen_id = c.citizen_id
WHERE EXTRACT(YEAR FROM v.date) = 2024
  AND c.educational_qualification = '10th class'
  AND c.age <= 18;
```

Task I: Total number of births of boy children in the year 2024

```sql
SELECT COUNT(*) AS total_boy_births
FROM citizens c
WHERE gender = 'Male'
  AND EXTRACT(YEAR FROM c.date_of_birth) = 2024;
```

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

```sql
SELECT COUNT(DISTINCT c.citizen_id) AS total_citizens
FROM citizens c
WHERE c.household_id IN (
    SELECT DISTINCT p.household_id
    FROM panchayat_employees p
    JOIN citizens pc ON p.citizen_id = pc.citizen_id
);
```