**Part 3- SQL scripts**

**Creating the schema using the SQL statements**

CREATE DATABASE CleanWaterAndSanitationDB;

USE CleanWaterAndSanitationDB;

CREATE TABLE Communities (

Community\_ID INT PRIMARY KEY AUTO\_INCREMENT,

Name VARCHAR(100) NOT NULL,

Location VARCHAR(150) NOT NULL,

Population INT NOT NULL

);

CREATE TABLE Water\_Sources (

Source\_ID INT PRIMARY KEY AUTO\_INCREMENT,

Source\_Type VARCHAR(50) NOT NULL,

Location VARCHAR(150) NOT NULL,

Community\_ID INT,

FOREIGN KEY (Community\_ID) REFERENCES Communities(Community\_ID)

);

CREATE TABLE Water\_Quality\_Tests (

Test\_ID INT PRIMARY KEY AUTO\_INCREMENT,

Source\_ID INT,

Test\_Date DATE NOT NULL,

Contaminants\_Level DECIMAL(5, 2),

Potability\_Status VARCHAR(50),

FOREIGN KEY (Source\_ID) REFERENCES Water\_Sources(Source\_ID)

);

CREATE TABLE Projects (

Project\_ID INT PRIMARY KEY AUTO\_INCREMENT,

Project\_Name VARCHAR(100) NOT NULL,

Start\_Date DATE NOT NULL,

End\_Date DATE,

Community\_ID INT,

FOREIGN KEY (Community\_ID) REFERENCES Communities(Community\_ID)

);

CREATE TABLE Project\_Funding (

Funding\_ID INT PRIMARY KEY AUTO\_INCREMENT,

Project\_ID INT,

Funder\_Name VARCHAR(100) NOT NULL,

Amount DECIMAL(10, 2) NOT NULL,

FOREIGN KEY (Project\_ID) REFERENCES Projects(Project\_ID)

);

CREATE TABLE Beneficiaries (

Beneficiary\_ID INT PRIMARY KEY AUTO\_INCREMENT,

Name VARCHAR(100) NOT NULL,

Household\_Size INT NOT NULL,

Community\_ID INT,

FOREIGN KEY (Community\_ID) REFERENCES Communities(Community\_ID)

);

**Populating the database tables.**

INSERT INTO Communities (Name, Location, Population)

VALUES

('Village A', 'Region 1', 1200),

('Village B', 'Region 2', 800),

('Village C', 'Region 3', 1500);

|  |  |  |  |
| --- | --- | --- | --- |
| **Community\_ID** | **Name** | **Location** | **Population** |
| 1 | Village A | Region 1 | 1200 |
| 2 | Village B | Region 2 | 800 |
| 3 | Village C | Region 3 | 1500 |

INSERT INTO Water\_Sources (Source\_Type, Location, Community\_ID)

VALUES

('Well', 'Region 1', 1),

('River', 'Region 2', 2),

('Borehole', 'Region 3', 3);

|  |  |  |  |
| --- | --- | --- | --- |
| **Source\_ID** | **Source\_Type** | **Location** | **Community\_ID** |
| 1 | Well | Region 1 | 1 |
| 2 | River | Region 2 | 2 |
| 3 | Borehole | Region 3 | 3 |

INSERT INTO Water\_Quality\_Tests (Source\_ID, Test\_Date, Contaminants\_Level, Potability\_Status)

VALUES

(1, '2024-01-15', 2.5, 'Safe'),

(2, '2024-02-20', 10.0, 'Unsafe'),

(3, '2024-03-10', 1.2, 'Safe');

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test\_ID** | **Source\_ID** | **Test\_Date** | **Contaminants\_Level** | **Potability\_Status** |
| 1 | 1 | 2024-01-15 | 2.5 | Safe |
| 2 | 2 | 2024-02-20 | 10.0 | Unsafe |
| 3 | 3 | 2024-03-10 | 1.2 | Safe |

INSERT INTO Projects (Project\_Name, Start\_Date, End\_Date, Community\_ID)

VALUES

('Well Improvement Project', '2024-01-01', '2024-06-30', 1),

('River Water Treatment', '2024-02-01', '2024-07-31', 2),

('Borehole Drilling', '2024-03-01', '2024-08-31', 3);

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Project\_ID** | **Project\_Name** | **Start\_Date** | **End\_Date** | **Community\_ID** |
| 1 | Well Improvement Project | 2024-01-01 | 2024-06-30 | 1 |
| 2 | River Water Treatment | 2024-02-01 | 2024-07-31 | 2 |
| 3 | Borehole Drilling | 2024-03-01 | 2024-08-31 | 3 |

INSERT INTO Project\_Funding (Project\_ID, Funder\_Name, Amount)

VALUES

(1, 'WaterAid', 50000.00),

(2, 'UNICEF', 75000.00),

(3, 'WHO', 100000.00);

|  |  |  |  |
| --- | --- | --- | --- |
| **Funding\_ID** | **Project\_ID** | **Funder\_Name** | **Amount** |
| 1 | 1 | WaterAid | 50000.00 |
| 2 | 2 | UNICEF | 75000.00 |
| 3 | 3 | WHO | 100000.00 |

INSERT INTO Beneficiaries (Name, Household\_Size, Community\_ID)

VALUES

('John Doe', 5, 1),

('Jane Smith', 3, 2),

('Michael Johnson', 6, 3);

|  |  |  |  |
| --- | --- | --- | --- |
| **Beneficiary\_ID** | **Name** | **Household\_Size** | **Community\_ID** |
| 1 | John Doe | 5 | 1 |
| 2 | Jane Smith | 3 | 2 |
| 3 | Michael Johnson | 6 | 3 |

**Part3 –SQL Programming**

**Retrieval**

In the SQL programming, we are going to retrieve and anayze the retreived data according or based on problem definition which includes; list of all communities and their population, Retrieve water sources and their potability status for each community,details of projects implemented in communities with their funding information and retrieving the list of beneficiaries in each community

1. **Retrieve a list of all communities and their population**

SELECT Community\_ID, Name AS Community\_Name, Population

FROM Communities;

**b. Retrieve water sources and their potability status for each community**

SELECT c.Name AS Community\_Name, ws.Source\_Type, ws.Location, wq.Potability\_Status

FROM Water\_Sources ws

JOIN Communities c ON ws.Community\_ID = c.Community\_ID

JOIN Water\_Quality\_Tests wq ON ws.Source\_ID = wq.Source\_ID;

**c. Retrieve details of projects implemented in communities with their funding information**

SELECT c.Name AS Community\_Name, p.Project\_Name, p.Start\_Date, p.End\_Date, pf.Funder\_Name, pf.Amount

FROM Projects p

JOIN Communities c ON p.Community\_ID = c.Community\_ID

JOIN Project\_Funding pf ON p.Project\_ID = pf.Project\_ID;

**d. Retrieve the list of beneficiaries in each community**

SELECT b.Name AS Beneficiary\_Name, c.Name AS Community\_Name, b.Household\_Size

FROM Beneficiaries b

JOIN Communities c ON b.Community\_ID = c.Community\_ID;

**Data Analysis Queries**

**a. Analyze the potability status of water sources across all communities**

The query will help identify the number of safe and unsafe water sources, providing insight into the overall water quality in the communities.

SELECT wq.Potability\_Status, COUNT(ws.Source\_ID) AS Number\_Of\_Water\_Sources

FROM Water\_Quality\_Tests wq

JOIN Water\_Sources ws ON wq.Source\_ID = ws.Source\_ID

GROUP BY wq.Potability\_Status;

**b. Calculate the average contaminants level in water sources by community**

SELECT c.Name AS Community\_Name, AVG(wq.Contaminants\_Level) AS Average\_Contaminants\_Level

FROM Water\_Quality\_Tests wq

JOIN Water\_Sources ws ON wq.Source\_ID = ws.Source\_ID

JOIN Communities c ON ws.Community\_ID = c.Community\_ID

GROUP BY c.Name;

This query will reveal which communities have higher average contaminants levels in their water sources, indicating areas that may require urgent intervention.

1. **Retrieve the total funding received for projects in each community**

The query will show how much funding each community has received, helping to assess if there is a correlation between funding and improvements in water quality.

SELECT c.Name AS Community\_Name, SUM(pf.Amount) AS Total\_Funding

FROM Project\_Funding pf

JOIN Projects p ON pf.Project\_ID = p.Project\_ID

JOIN Communities c ON p.Community\_ID = c.Community\_ID

GROUP BY c.Name;

**d. Identify communities with the highest number of beneficiaries**

Here, we are to identify which communities have the largest number of beneficiaries, which could correlate with higher population needs or more extensive project implementations.

SELECT c.Name AS Community\_Name, COUNT(b.Beneficiary\_ID) AS Number\_Of\_Beneficiaries

FROM Beneficiaries b

JOIN Communities c ON b.Community\_ID = c.Community\_ID

GROUP BY c.Name

ORDER BY Number\_Of\_Beneficiaries DESC;

**e. Determine the success of water quality improvement projects by comparing water quality before and after project implementation:**

SELECT c.Name AS Community\_Name, ws.Source\_Type, MIN(wq.Test\_Date) AS Earliest\_Test\_Date,

MAX(wq.Test\_Date) AS Latest\_Test\_Date,

(SELECT AVG(wq1.Contaminants\_Level)

FROM Water\_Quality\_Tests wq1

WHERE wq1.Source\_ID = ws.Source\_ID

AND wq1.Test\_Date = MIN(wq.Test\_Date)) AS Initial\_Contaminants\_Level,

(SELECT AVG(wq2.Contaminants\_Level)

FROM Water\_Quality\_Tests wq2

WHERE wq2.Source\_ID = ws.Source\_ID

AND wq2.Test\_Date = MAX(wq.Test\_Date)) AS Final\_Contaminants\_Level

FROM Water\_Quality\_Tests wq

JOIN Water\_Sources ws ON wq.Source\_ID = ws.Source\_ID

JOIN Communities c ON ws.Community\_ID = c.Community\_ID

GROUP BY ws.Source\_ID, c.Name, ws.Source\_Type;

This query will compare the contaminants level before and after project implementation, providing insight into the effectiveness of the interventions.