**MSIS537-DATA MANAGEMENT 1**

Final project report

**Drinking Water Distribution Management**

Swetha Adike

CWID -20133856

[1. ABSTRACT 3](#_Toc90658096)

[2. OBJECTIVE 3](#_Toc90658097)

[3. EXECUTIVE SUMMARY 4](#_Toc90658098)

[4. ENTITY RELATIONSHIP (ER) DIAGRAM 5](#_Toc90658099)

[5. TABLES 6](#_Toc90658100)

[a. District Table 6](#_Toc90658101)

[b. Mandal Table 7](#_Toc90658102)

[c. Habitation Table 8](#_Toc90658103)

[d. Source Table 10](#_Toc90658104)

[e. Structure Table 11](#_Toc90658105)

[f. Service Reservoirs Table 13](#_Toc90658106)

[g. Household Connections Table 15](#_Toc90658107)

[h. Network Table 16](#_Toc90658108)

[6. Queries and Output 18](#_Toc90658109)

[7. IMPLEMENTATION NOTES 21](#_Toc90658110)

[8. KNOWN PROBLEMS 21](#_Toc90658111)

[9. FUTURE ENHANCEMENTS 22](#_Toc90658112)

# ABSTRACT

Water distribution system consists of an infrastructure that collect, treat, store and distribute water between sources and households. It is a combination of water sources, intake structures and service reservoirs through which water is distributed to consumers by a web of pipeline. To make it understandable, each component of water channels will be explained. The water distribution process consists of a huge database where there is a need to store data of each component on a daily basis or twice in a day, depending on the usage of water by customers. In this project, a logical database is created and engineered to relational data base and cardinalities are drawn between all the tables. Data is then inserted into these tables after importing the DDL file from Oracle SQL modeler to Oracle SQL Developer. Testing is also done using few SQL query commands to retrieve the data.

# OBJECTIVE

The main objective of the project is to develop a relational database for drinking water distribution system which stores and keeps track of the information regarding the water sources, distribution structures, service reservoirs, household connections and pipelines in a state using Oracle SQL Modeler and Oracle SQL Developer. Aim of this project is to retrieve the information regarding the water sources in a habitation and also get information of network system, like from which structure water is distributed to a particular habitation or a household connection in a habitation. Also to draw information regarding the pipelines, in which a service reservoir is serving in a habitation

# EXECUTIVE SUMMARY

Water distribution system carries water from a centralized water sources to a treatment plant. The portable water is then supplied to the consumers after treatment through network of pipes. The information regarding the water availability in each water distribution system component and supply to the households is maintained in digital format and physical format in each area, for day to day reference to release water to the consumers. This information in digital format is maintained either in SCADA system or in online or both.

Water distribution management, in this project, is a database management which refers to all water distribution system frameworks in a state, under taken by a government allied department, Mission Bhagiratha Department. The information is monitored and utilized by officers in the department at various levels to enhance the maintenance operations of all structures. The database consists of various water distribution system components, used in this project, in the order of the deliver water to a household are: Segments, Sources, Structures, and Service reservoirs.

Before going into the definitions of these components, there are few geographical divisions by which a State on a whole is divided. Also there are few terms which are to be explained to understand this project better, they are: District, Mandal, and Habitation. A State is divided into few **Districts** and these districts are subdivided into **Mandals**. Each Mandal is a group of **Habitations** put together. Each habitation consists of group of **households** which are customers, to which water is supplied by pipeline **network**. A household connection is a tap connection given to that household.

**Segment:** Depending upon the water availability, and irrespective of districts, a State is again divided into few segments. The division of segments is such that the geographical area fed by a particular river or lake comes under one segment. This helped the project in reducing the cost on the overall project.

**Sources**: Each river or lake watered to a segment is denoted as a source.

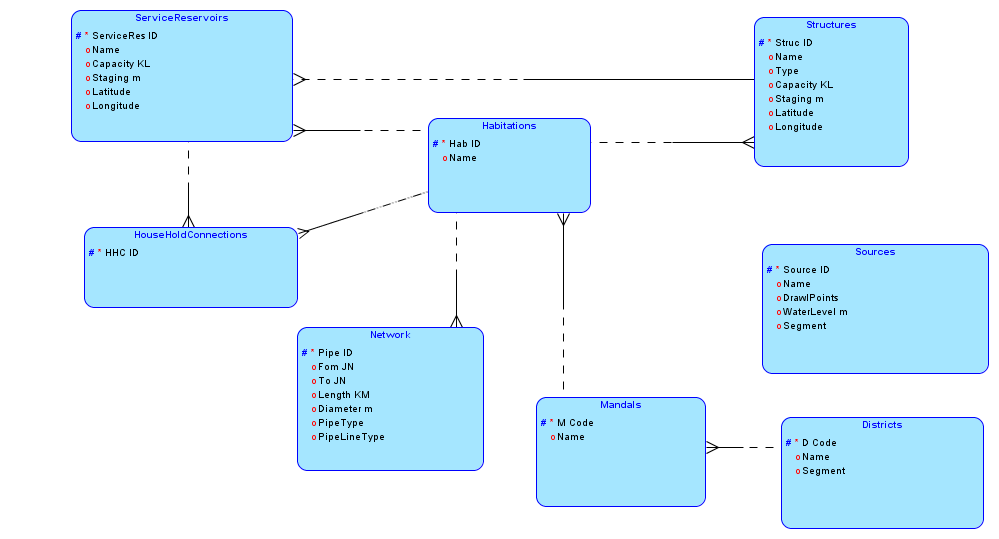
**Structure**: The water from source has to be stored temporarily in a structure for treating the water. The structure component has various types like intake wells, water treatment plants, break pressure tanks, and over head balancing reservoirs. Each of these is a temporary storage house for water to be treated or for treated water.

**Service Reservoir**: Water from these structures is stored in these service reservoirs or over head tanks, which is the water ready for supplying.

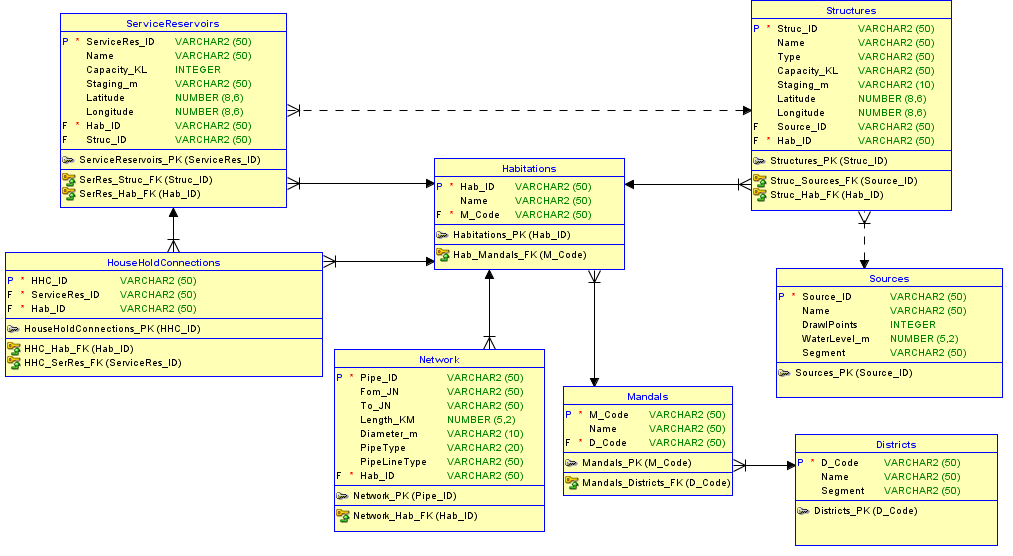
After relating all the above explained components in tables with respective fields, the cardinalities will explain the flow of the data in this database. The database also includes latitude and longitude to locate a structure component. By creating this database, quick information of structures and its allied service reservoir information can be drawn. Also, the relationship of a structure with a pipeline network can be extracted. All the possible complex queries were shown as an output to query in section 6.

# ENTITY RELATIONSHIP (ER) DIAGRAM

The table entities are first created in logical data model in Oracle SQL data modeler as the figure below:



Then the above are engineered to relational data model in Oracle SQL data modeler as the figure below



# TABLES

First, a table is created in the Oracle SQL data modeler and the fields are entered with their respective data types. After creating all tables, cardinal relationships are developed between the tables depending upon functional dependency. The details of the tables in above ER diagram and their cardinalities are given below.

## District Table

1. **Create** **Statement**

CREATE TABLE districts (

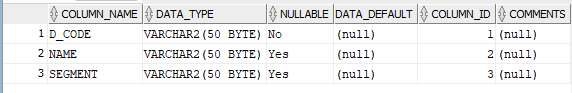
d\_code VARCHAR2(50) NOT NULL,

name VARCHAR2(50),

segment VARCHAR2(50)

);

1. **Table Result**



ALTER TABLE districts ADD CONSTRAINT districts\_pk PRIMARY KEY ( d\_code );

1. **Data Insertion**

INSERT INTO "DISTRICTS" (D\_CODE, NAME, SEGMENT) VALUES ('D1', 'District1', 'Segment1')INSERT INTO "DISTRICTS" (D\_CODE, NAME, SEGMENT) VALUES ('D2', 'District2', 'Segment2')

INSERT INTO "DISTRICTS" (D\_CODE, NAME, SEGMENT) VALUES ('D3', 'District3', 'Segment3')

INSERT INTO "DISTRICTS" (D\_CODE, NAME, SEGMENT) VALUES ('D4', 'District4', 'Segment4')

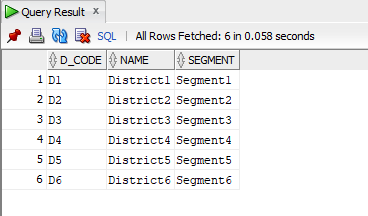
INSERT INTO "DISTRICTS" (D\_CODE, NAME, SEGMENT) VALUES ('D5', 'District5', 'Segment5')

INSERT INTO "DISTRICTS" (D\_CODE, NAME, SEGMENT) VALUES ('D6', 'District6', 'Segment6')

1. **Functional Dependency**

D\_Code 🡪Name, Segment

1. **Sample Data**

****

## Mandal Table

1. **Create Statement**

CREATE TABLE mandals (

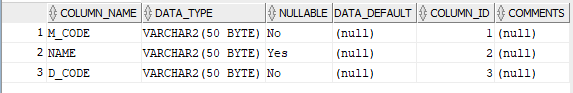
m\_code VARCHAR2(50) NOT NULL,

name VARCHAR2(50),

d\_code VARCHAR2(50) NOT NULL

);

1. **Table Result**

****

ALTER TABLE mandals ADD CONSTRAINT mandals\_pk PRIMARY KEY ( m\_code );

ALTER TABLE mandals

ADD CONSTRAINT mandals\_districts\_fk FOREIGN KEY ( d\_code )

REFERENCES districts ( d\_code );

1. **Functional Dependency**

M\_Code 🡪Name

1. **Data Insertion**

INSERT INTO "MANDALS" (M\_CODE, NAME, D\_CODE) VALUES ('M1', 'Mandal1', 'D1')INSERT INTO "MANDALS" (M\_CODE, NAME, D\_CODE) VALUES ('M2', 'Mandal2', 'D1')

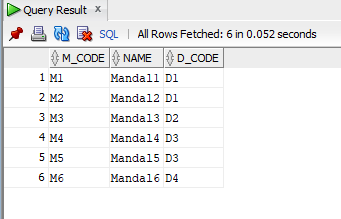
INSERT INTO "MANDALS" (M\_CODE, NAME, D\_CODE) VALUES ('M3', 'Mandal3', 'D2')

INSERT INTO "MANDALS" (M\_CODE, NAME, D\_CODE) VALUES ('M4', 'Mandal4', 'D3')

INSERT INTO "MANDALS" (M\_CODE, NAME, D\_CODE) VALUES ('M5', 'Mandal5', 'D3')

INSERT INTO "MANDALS" (M\_CODE, NAME, D\_CODE) VALUES ('M6', 'Mandal6', 'D4')

1. **Sample Data**

****

## Habitation Table

1. **Create Statement**

CREATE TABLE habitations (

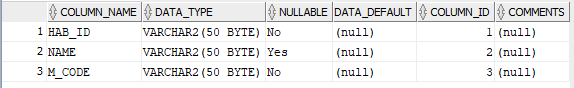
hab\_id VARCHAR2(50) NOT NULL,

name VARCHAR2(50),

m\_code VARCHAR2(50) NOT NULL

);

1. **Table Result**

****

ALTER TABLE habitations ADD CONSTRAINT habitations\_pk PRIMARY KEY ( hab\_id );

ALTER TABLE habitations

ADD CONSTRAINT hab\_mandals\_fk FOREIGN KEY ( m\_code )

REFERENCES mandals ( m\_code );

1. **Functional Dependency**

Hab\_ID🡪Name

1. **Data Insertion**

INSERT INTO "HABITATIONS" (HAB\_ID, NAME, M\_CODE) VALUES ('H1', 'Habitation1', 'M1')

INSERT INTO "HABITATIONS" (HAB\_ID, NAME, M\_CODE) VALUES ('H2', 'Habitation2', 'M1')

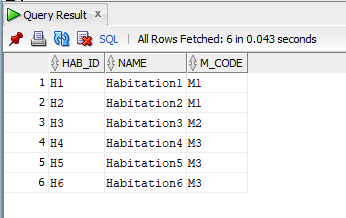
INSERT INTO "HABITATIONS" (HAB\_ID, NAME, M\_CODE) VALUES ('H3', 'Habitation3', 'M2')

INSERT INTO "HABITATIONS" (HAB\_ID, NAME, M\_CODE) VALUES ('H4', 'Habitation4', 'M3')

INSERT INTO "HABITATIONS" (HAB\_ID, NAME, M\_CODE) VALUES ('H5', 'Habitation5', 'M3')

INSERT INTO "HABITATIONS" (HAB\_ID, NAME, M\_CODE) VALUES ('H6', 'Habitation6', 'M3')

1. **Sample Data**

****

## Source Table

1. **Create Statement**

CREATE TABLE sources (

source\_id VARCHAR2(50) NOT NULL,

name VARCHAR2(50),

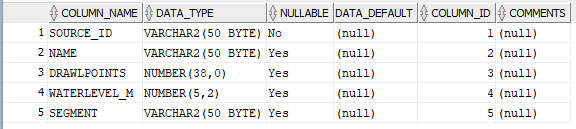
drawlpoints INTEGER,

waterlevel\_m NUMBER(5, 2),

segment VARCHAR2(50)

);

1. **Table Result**



ALTER TABLE sources ADD CONSTRAINT sources\_pk PRIMARY KEY ( source\_id );

1. **Functional Dependency**

Source\_ID🡪Name, DrawlPoints, WaterLevel\_m, Segment

1. **Sample Data Insertion**

INSERT INTO "SOURCES" (SOURCE\_ID, NAME, DRAWLPOINTS, WATERLEVEL\_M, SEGMENT) VALUES ('S1', 'Source1', '2', '128.56', 'Segment1')INSERT INTO "SOURCES" (SOURCE\_ID, NAME, DRAWLPOINTS, WATERLEVEL\_M, SEGMENT) VALUES ('S2', 'Source2', '1', '60.25', 'Segment2')

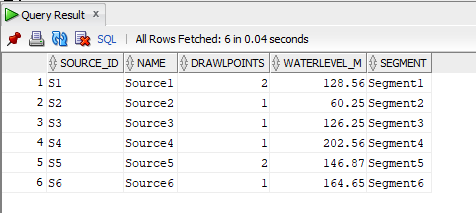
INSERT INTO "SOURCES" (SOURCE\_ID, NAME, DRAWLPOINTS, WATERLEVEL\_M, SEGMENT) VALUES ('S3', 'Source3', '1', '126.25', 'Segment3')

INSERT INTO "SOURCES" (SOURCE\_ID, NAME, DRAWLPOINTS, WATERLEVEL\_M, SEGMENT) VALUES ('S4', 'Source4', '1', '202.56', 'Segment4')

INSERT INTO "SOURCES" (SOURCE\_ID, NAME, DRAWLPOINTS, WATERLEVEL\_M, SEGMENT) VALUES ('S5', 'Source5', '2', '146.87', 'Segment5')

INSERT INTO "SOURCES" (SOURCE\_ID, NAME, DRAWLPOINTS, WATERLEVEL\_M, SEGMENT) VALUES ('S6', 'Source6', '1', '164.65', 'Segment6')

1. **Sample Data Table**

****

## Structure Table

1. **Create Statement**

CREATE TABLE structures (

struc\_id VARCHAR2(50) NOT NULL,

name VARCHAR2(50),

type VARCHAR2(50),

capacity\_kl VARCHAR2(50),

staging\_m VARCHAR2(10),

latitude NUMBER(8, 6),

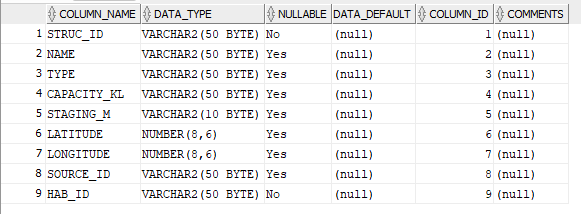
longitude NUMBER(8, 6),

source\_id VARCHAR2(50),

hab\_id VARCHAR2(50) NOT NULL

);

1. **Table Result**

****

ALTER TABLE structures ADD CONSTRAINT structures\_pk PRIMARY KEY ( struc\_id );

ALTER TABLE structures

ADD CONSTRAINT struc\_hab\_fk FOREIGN KEY ( hab\_id )

REFERENCES habitations ( hab\_id );

ALTER TABLE structures

ADD CONSTRAINT struc\_sources\_fk FOREIGN KEY ( source\_id )

REFERENCES sources ( source\_id );

1. **Functional Dependency**

Struc\_ID🡪Name, Type, Capacity\_KL, Staging\_m, Latitude, Longitude,

1. **Sample Data Insertion**

INSERT INTO "STRUCTURES" (STRUC\_ID, NAME, TYPE, CAPACITY\_KL, LATITUDE, LONGITUDE, SOURCE\_ID, HAB\_ID) VALUES ('S1', 'Structure1', 'Intake Well', '40X65 Sqm', '78.635874', '16.245789', 'S1', 'H1')INSERT INTO "STRUCTURES" (STRUC\_ID, NAME, TYPE, CAPACITY\_KL, LATITUDE, LONGITUDE, SOURCE\_ID, HAB\_ID) VALUES ('S2', 'Structure2', 'Water Treatment Plant', '230 MLD', '77.856987', '16.532986', 'S2', 'H4')

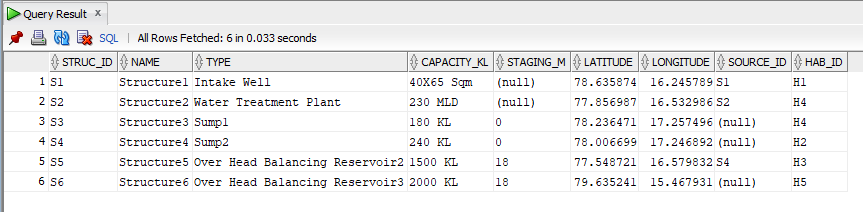
INSERT INTO "STRUCTURES" (STRUC\_ID, NAME, TYPE, CAPACITY\_KL, STAGING\_M, LATITUDE, LONGITUDE, HAB\_ID) VALUES ('S3', 'Structure3', 'Sump1', '180 KL', '0', '78.236471', '17.257496', 'H4')

INSERT INTO "STRUCTURES" (STRUC\_ID, NAME, TYPE, CAPACITY\_KL, STAGING\_M, LATITUDE, LONGITUDE, HAB\_ID) VALUES ('S4', 'Structure4', 'Sump2', '240 KL', '0', '78.006699', '17.246892', 'H2')

INSERT INTO "STRUCTURES" (STRUC\_ID, NAME, TYPE, CAPACITY\_KL, STAGING\_M, LATITUDE, LONGITUDE, SOURCE\_ID, HAB\_ID) VALUES ('S5', 'Structure5', 'Over Head Balancing Reservoir2', '1500 KL', '18', '77.548721', '16.579832', 'S4', 'H3')

INSERT INTO "STRUCTURES" (STRUC\_ID, NAME, TYPE, CAPACITY\_KL, STAGING\_M, LATITUDE, LONGITUDE, HAB\_ID) VALUES ('S6', 'Structure6', 'Over Head Balancing Reservoir3', '2000 KL', '18', '79.635241', '15.467931', 'H5')

1. **Sample Data Table**

****

## Service Reservoirs Table

1. **Create Statement**

CREATE TABLE servicereservoirs (

serviceres\_id VARCHAR2(50) NOT NULL,

name VARCHAR2(50),

capacity\_kl INTEGER,

staging\_m VARCHAR2(50),

latitude NUMBER(8, 6),

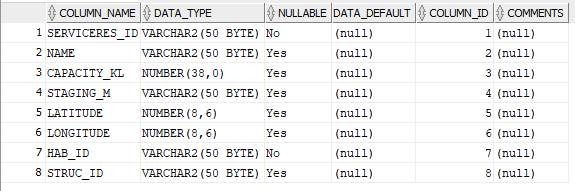
longitude NUMBER(8, 6),

hab\_id VARCHAR2(50) NOT NULL,

struc\_id VARCHAR2(50)

);

1. **Table Result**

****

ALTER TABLE servicereservoirs ADD CONSTRAINT servicereservoirs\_pk PRIMARY KEY ( serviceres\_id );

ALTER TABLE servicereservoirs

ADD CONSTRAINT serres\_hab\_fk FOREIGN KEY ( hab\_id )

REFERENCES habitations ( hab\_id );

ALTER TABLE servicereservoirs

ADD CONSTRAINT serres\_struc\_fk FOREIGN KEY ( struc\_id )

REFERENCES structures ( struc\_id );

1. **Functional Dependency**

ServiceRes\_ID🡪Name, Capacity\_KL, Staging\_m, Latitude, Longitude

1. **Sample Data Insertion**

INSERT INTO "SERVICERESERVOIRS" (SERVICERES\_ID, NAME, CAPACITY\_KL, STAGING\_M, HAB\_ID, STRUC\_ID) VALUES ('SR1', 'SerReservoir1', '90', '12', 'H1', 'S5')Insert into "SERVICERESERVOIRS" (LATITUDE, LONGITUDE) Values ('78.239874', '16.358416')

INSERT INTO "SERVICERESERVOIRS" (SERVICERES\_ID, NAME, CAPACITY\_KL, STAGING\_M, LATITUDE, LONGITUDE, HAB\_ID, STRUC\_ID) VALUES ('SR2', 'SerReservoir2', '10', '9', '78.124576', '16.856341', 'H1', 'S5')

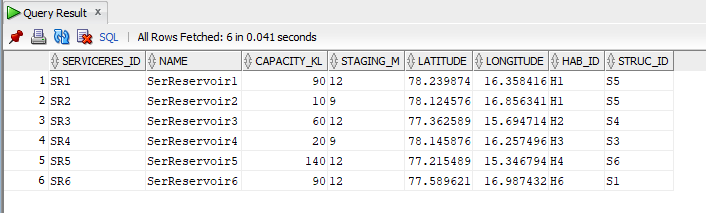
INSERT INTO "SERVICERESERVOIRS" (SERVICERES\_ID, NAME, CAPACITY\_KL, STAGING\_M, LATITUDE, LONGITUDE, HAB\_ID, STRUC\_ID) VALUES ('SR3', 'SerReservoir3', '60', '12', '77.362589', '15.694714', 'H2', 'S4')

INSERT INTO "SERVICERESERVOIRS" (SERVICERES\_ID, NAME, CAPACITY\_KL, STAGING\_M, LATITUDE, LONGITUDE, HAB\_ID, STRUC\_ID) VALUES ('SR4', 'SerReservoir4', '20', '9', '78.145876', '16.257496', 'H3', 'S3')

INSERT INTO "SERVICERESERVOIRS" (SERVICERES\_ID, NAME, CAPACITY\_KL, STAGING\_M, LATITUDE, LONGITUDE, HAB\_ID, STRUC\_ID) VALUES ('SR5', 'SerReservoir5', '140', '12', '77.215489', '15.346794', 'H4', 'S6')

INSERT INTO "SERVICERESERVOIRS" (SERVICERES\_ID, NAME, CAPACITY\_KL, STAGING\_M, LATITUDE, LONGITUDE, HAB\_ID, STRUC\_ID) VALUES ('SR6', 'SerReservoir6', '90', '12', '77.589621', '16.987432', 'H6', 'S1')

1. **Sample Data Table**

****

## Household Connections Table

1. **Create Statement**

CREATE TABLE householdconnections (

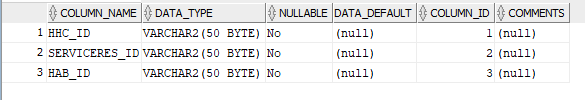
hhc\_id VARCHAR2(50) NOT NULL,

serviceres\_id VARCHAR2(50) NOT NULL,

hab\_id VARCHAR2(50) NOT NULL

);

1. **Table result**



ALTER TABLE householdconnections ADD CONSTRAINT householdconnections\_pk PRIMARY KEY ( hhc\_id );

ALTER TABLE householdconnections

ADD CONSTRAINT hhc\_hab\_fk FOREIGN KEY ( hab\_id )

REFERENCES habitations ( hab\_id );

ALTER TABLE householdconnections

ADD CONSTRAINT hhc\_serres\_fk FOREIGN KEY ( serviceres\_id )

REFERENCES servicereservoirs ( serviceres\_id );

1. **Sample Data Insertion**

INSERT INTO "HOUSEHOLDCONNECTIONS" (HHC\_ID, SERVICERES\_ID, HAB\_ID) VALUES ('HHC1', 'SR1', 'H1')

INSERT INTO "HOUSEHOLDCONNECTIONS" (HHC\_ID, SERVICERES\_ID, HAB\_ID) VALUES ('HHC2', 'SR1', 'H1')

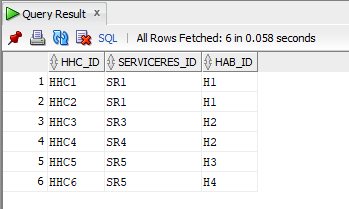
INSERT INTO "HOUSEHOLDCONNECTIONS" (HHC\_ID, SERVICERES\_ID, HAB\_ID) VALUES ('HHC3', 'SR3', 'H2')

INSERT INTO "HOUSEHOLDCONNECTIONS" (HHC\_ID, SERVICERES\_ID, HAB\_ID) VALUES ('HHC4', 'SR4', 'H2')

INSERT INTO "HOUSEHOLDCONNECTIONS" (HHC\_ID, SERVICERES\_ID, HAB\_ID) VALUES ('HHC5', 'SR5', 'H3')

INSERT INTO "HOUSEHOLDCONNECTIONS" (HHC\_ID, SERVICERES\_ID, HAB\_ID) VALUES ('HHC6', 'SR5', 'H4')

1. **Sample Data Table**

****

## Network Table

1. **Create Statement**

CREATE TABLE network (

pipe\_id VARCHAR2(50) NOT NULL,

fom\_jn VARCHAR2(50),

to\_jn VARCHAR2(50),

length\_km NUMBER(5, 2),

diameter\_m VARCHAR2(10),

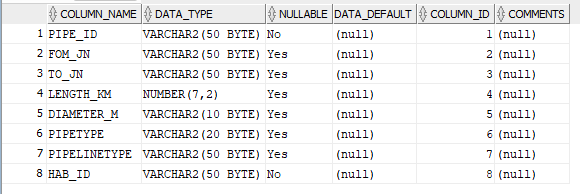
pipetype VARCHAR2(20),

pipelinetype VARCHAR2(50),

hab\_id VARCHAR2(50) NOT NULL

);

1. **Table Result**

****

ALTER TABLE network ADD CONSTRAINT network\_pk PRIMARY KEY ( pipe\_id );

ALTER TABLE network

ADD CONSTRAINT network\_hab\_fk FOREIGN KEY ( hab\_id )

REFERENCES habitations ( hab\_id );

1. **Functional Dependency**

Pipe\_ID🡪From\_JN, To\_JN, Length\_KM, Diameter\_m, PipeType, PipeLineType

1. **Sample Data Insertion**

INSERT INTO "NETWORK" (PIPE\_ID, FOM\_JN, TO\_JN, LENGTH\_KM, DIAMETER\_M, PIPETYPE, PIPELINETYPE, HAB\_ID) VALUES ('P1', 'JN1', 'JN2', '2.25', '56.6', 'DI', 'Gravity Main', 'H1')

INSERT INTO "NETWORK" (PIPE\_ID, FOM\_JN, TO\_JN, LENGTH\_KM, DIAMETER\_M, PIPETYPE, PIPELINETYPE, HAB\_ID) VALUES ('P2', 'JN2', 'JN3', '1.18', '73.2', 'PVC', 'Gravity Main', 'H3')

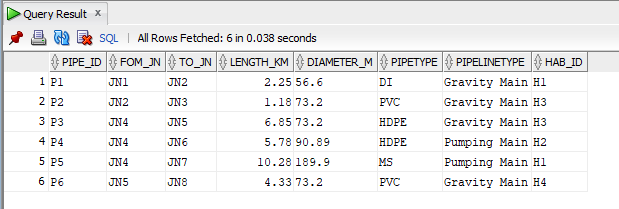
INSERT INTO "NETWORK" (PIPE\_ID, FOM\_JN, TO\_JN, LENGTH\_KM, DIAMETER\_M, PIPETYPE, PIPELINETYPE, HAB\_ID) VALUES ('P3', 'JN4', 'JN5', '6.85', '73.2', 'HDPE', 'Gravity Main', 'H3')

INSERT INTO "NETWORK" (PIPE\_ID, FOM\_JN, TO\_JN, LENGTH\_KM, DIAMETER\_M, PIPETYPE, PIPELINETYPE, HAB\_ID) VALUES ('P4', 'JN4', 'JN6', '5.78', '90.89', 'HDPE', 'Pumping Main', 'H2')

INSERT INTO "NETWORK" (PIPE\_ID, FOM\_JN, TO\_JN, LENGTH\_KM, DIAMETER\_M, PIPETYPE, PIPELINETYPE, HAB\_ID) VALUES ('P5', 'JN4', 'JN7', '10.28', '189.9', 'MS', 'Pumping Main', 'H1')

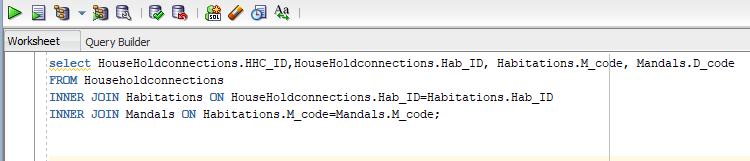
INSERT INTO "NETWORK" (PIPE\_ID, FOM\_JN, TO\_JN, LENGTH\_KM, DIAMETER\_M, PIPETYPE, PIPELINETYPE, HAB\_ID) VALUES ('P6', 'JN5', 'JN8', '4.33', '73.2', 'pvc', 'Gravity Main', 'H4')

1. **Sample Data Table**

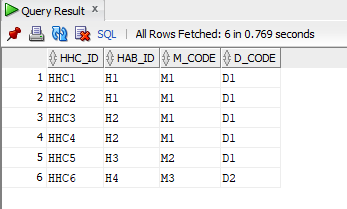
****

# Queries and Output

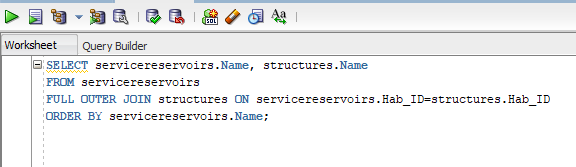
**Output to query 1**: To get household connection location in a state



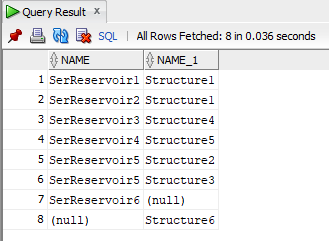
And the result is



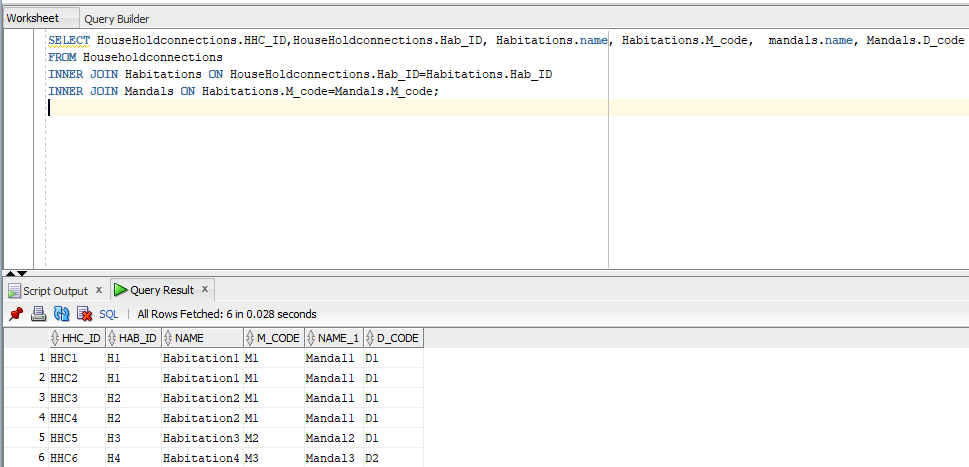
**Output to query 2**: To get which structures are watered by the sources

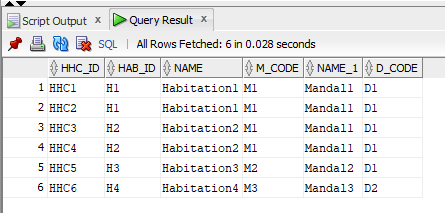


And the result is

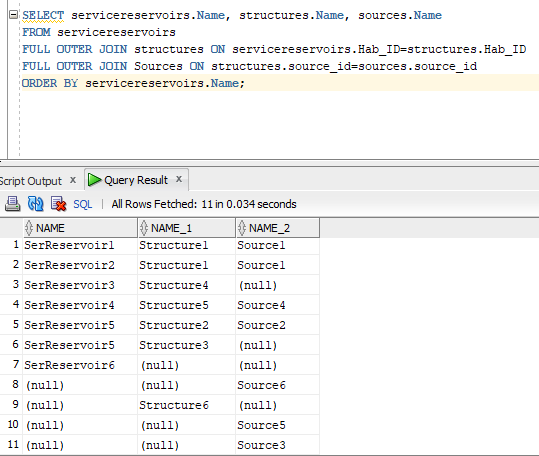


**Output to query 3**: To get habitation and Mandal IDs & Names along with District IDs for household connections.

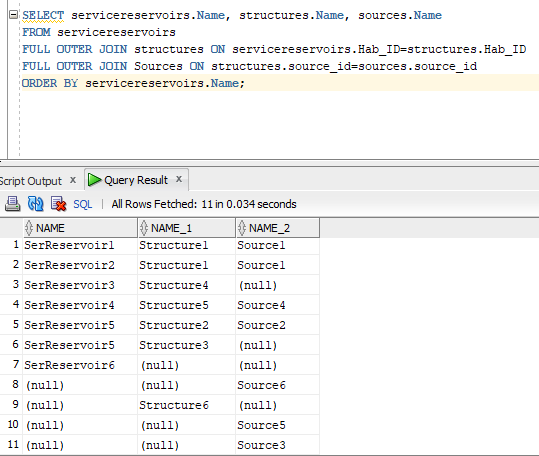




**Output to query 4:** To find out the connections of all structures and sources with service reservoirs



And the result is



# IMPLEMENTATION NOTES

November 10th, 2021 – Decided the topic of the project and worked on rough ER Diagram

November 18th, 2021 – Created Sources, District, Mandals and Habitations Tables in SQL Data

Modeler

November 22nd, 2021 – Worked on creating structures, service reservoirs tables

November 25th, 2021 – Designed ER Diagram for Water Distribution System Management in

SQL Data Modeler

December 4th, 2021 – Concluded with ER Diagram with cardinalities and catalogued all

required data into Excel Spreadsheet

December 7th, 2021 – Imported DDL statements to Oracle SQL Developer

December 10th, 2021 – Started inserting data into the tables which are already created

December 12th, 2021 – Sample data insertion was done

December 15th, 2021– Created SQL queries and ran them in SQL Data Developer

December 16th, 2021 – Worked on documentation and prepared for presenting the project in

class

December 17th, 2021 – Worked on future enhancements and known problems and finalized the

report to be submitted

# KNOWN PROBLEMS

1. This database is currently designed for limited pipeline networks. It can be upgraded or modified to accommodate different types of pipelines like gravity main and pumping main, one in each pipeline network table.
2. Segment data field in this database is duplicated in two tables. But this can be eliminated either by inserting more fields in the tables or by creating new cardinalities.
3. In real time, every structure should have a habitation connected to it but in query 2, there are few unconnected structures.

# FUTURE ENHANCEMENTS

1. The network table can be upgraded to accommodate little other type of pipelines too.
2. By segregating all the structures, service reservoirs and household connections with latitude, latitude and by denoting them with a district, mandal and habitation code, query can be drawn to extract a structure by its geographical location.
3. In pipe line network, sometimes the from\_JN can be one habitation and to\_JN can be in another. This can be found extended the network table to from\_JNHab and to\_JNHab, and relate this table to habitations table.
4. The database can be extended to water refilling and water billing systems.
5. The household connections table can be extended upgrade customer information.
6. Also the database must be upgraded in a timely manner patches may be needed to enhance the quality of output.