**Term Project Final Report**

**Electricity Management System** (EMS)

**INFO 5707 - Data Modeling for Information Professionals**

**Prof: Abiodun Awojobi**

 

**Group 5 - Eagles**

Pavan kalyan Naini

Mohith Sai Yandra

Sushanth Reddy Marreddy

Sharathchandra Reddy Yedavelli

Harshavardhan Reddy Basani

**CONTENTS**

**Project Description 3**

**Objectives 3**

**Scope 4**

**Project Requirements 4**

**Database Requirements 4**

**User Requirements 4**

**Business Rules 5**

**Entity Relationship Diagram 6**

**Data Dictionary 6**

**Queries and Operations 11**

**Section 1: Entity Generation and Data Entry 11**

**Overview 11**

**Entity Generation and Data Entry for Table Service\_Provider 12**

**Entity Generation and Data Entry for Table Region 14**

**Entity Generation and Data Entry for Table Customers 16**

**Entity Generation and Data Entry for Table Maintenance\_Requests 20**

**Entity Generation and Data Entry for Table Meter\_Readings 22**

**Entity Generation and Data Entry for Table Auto\_Payments 25**

**Entity Generation and Data Entry for Table Billing 26**

**Entity Generation and Data Entry for Table Reminders 29**

**Entity Generation and Data Entry for Table Usage\_History 32**

**Section 2: Data Retrieval and Analytical Reports 35**

**Overview 35**

**Data Analysis 1 35**

**Data Analysis 2 36**

**Data Analysis 3 37**

**Data Analysis 4 38**

**Data Analysis 5 39**

**Data Analysis 6 40**

**Data Analysis 7 41**

**Data Analysis 8 42**

**Data Analysis 9 43**

**Data Analysis 10 44**

**Appendix 1 46**

**Project Description**

The Electricity Management System represents a groundbreaking database initiative crafted to optimize the oversight of electricity consumption and distribution within a state. This sophisticated system serves as a central repository for crucial information, catering to the needs of both electricity authorities and consumers. It offers real-time monitoring capabilities for authorities, allowing them to track consumption patterns, device specifics, and billing information. This facilitates proactive decision-making and ensures the prompt identification and resolution of potential issues. The system's meticulous record-keeping feature is essential, securely capturing transaction details and safeguarding sensitive customer information.

Customers, in turn, experience enhanced convenience through a user-friendly interface that provides access to vital information. They can effortlessly track their monthly electricity usage, review payment history, stay informed about payment deadlines, and receive updates on potential outages. This transparency not only empowers consumers but also fosters responsible electricity usage. The system's integration of data from multiple electricity providers into a unified database is a cornerstone of its functionality. This integrated approach ensures the accuracy and consistency of information across diverse elements such as customer data, electricity bills, and transaction details, reinforcing the reliability and effectiveness of the state's electricity infrastructure.

In summary, the Electricity Management System is a comprehensive and user-centric solution designed to revolutionize electricity consumption and distribution management and monitoring. Its advanced features strengthen the operational efficiency of authorities, enhance customer satisfaction, and promote responsible energy consumption practices, thereby contributing to the overall reliability and transparency of the state's electricity infrastructure.

**Objectives**

* **Customer Information:**  The system will store and manage detailed customer information, including personal details, contact information, and any relevant preferences.
* **Transaction History:**  The system will maintain a record of all customer transactions, including payment history, date and time of transactions, and the corresponding amounts.
* **Electricity Usage Details:**  It will store and display historical data on electricity usage for each customer to analyze consumption patterns and make informed decisions.
* **Date-Specific Usage:**  The system will allow users to view electricity usage on specific dates. This feature is useful for both customers and authorities to understand consumption patterns on respective days.
* **Distributor Information:**  Electricity distributors will have the information stored in the system to ensure that authorities can manage and track distributor-related data efficiently.
* **Employee Information:** All information related to employees involved in the electricity management process will be stored, including details about their roles, responsibilities, and contact information.
* **User Preferences for Automatic Payments:**  The system will keep track of customer preferences, particularly those who have opted for automatic payments to ensure seamless billing processes for these users.

**Scope**

* **Electricity Usage Monitoring:** The system provides a platform for monitoring electricity usage, allowing authorities and customers to track consumption patterns.
* **Payment Details:** Customers can access and review their payment details, including transaction history and automatic payment preferences.
* **Comprehensive Customer Database:** A centralized database for customer information simplifies management tasks and enhances customer service.
* **Data Analysis for Decision-Making:** Previous data is analyzed to make informed decisions about electricity distribution and resource allocation.

**Project Requirements**

**Operating System:** Windows/ MacOs

**Database:** MSSQL

**Applications:** Azure Data Studio, Lucid chart, Microsoft word

**Database Requirements**

The following is the list that provides different table names that will be used.

1. Service\_Provider Table
2. Region Table
3. Customers Table
4. Maintenance\_Requests Table
5. Meter\_Readings Table
6. Auto\_Payments Table
7. Billing
8. Reminders Table
9. Usage\_History

**User Requirements**

* Usage Monitoring: The system should be able to track and monitor electricity usage for different periods (daily, monthly, and annually) for both residential and commercial purposes.
* Real-Time Data: Users should have access to real-time consumption data to monitor and adjust their usage as necessary.
* Billing Information: The system should allow users to view and manage their billing information, including past bills and payment history.
* Alerts and Notifications: Users should receive alerts for unusual consumption patterns, power outages, or billing notifications.
* Usage Insights: The system should provide analysis and insights on usage trends to help users optimize their consumption and reduce costs.
* User Management: The system should allow admin/manager users to add, remove, or modify user access and permissions.
* System Maintenance: Tools for system maintenance should be included, such as meter diagnostics, updates, and data integrity checks.
* Billing and Invoicing: Admin/manager users should be able to manage billing cycles, generate invoices, and handle payment processing.
* Compliance and Regulations: The system should ensure compliance with energy regulations and standards, with tools for reporting and adherence.
* Scalability: The system should be able to scale to accommodate a growing number of users or meters.
* Data Security: Robust security measures should be in place to protect user data and prevent unauthorized access.
* Data Backup and Recovery: Regular backups and a plan for data recovery in case of system failures or data loss.
* Compatibility and Integration: The system should be compatible with different metering systems and protocols for seamless integration.

**Business Rules**

1. Customer\_Id, Billing\_Id, Meter\_Id should be unique.
2. Each customer can have only one service provider.
3. Each customer should have one meter for billing.
4. A service provider can have many customers.
5. Each customer may have many usage history records and billing records.
6. Each meter reading should be associated with a unique Meter\_Id.
7. An auto-payment record ID or any payment method is associated with one customer and one bill.
8. Maintenancerequest\_Id should be specific to one meter and one customer, and some customers may not have maintenancerequest\_Id.
9. A particular region may have multiple customers and multiple service providers.
10. Each customer can be located in one region.
11. The total amount calculated for the bill should be based on the particular consumption period and the due balance from previous records.
12. The total payable amount should be equal to or greater than the bill amount generated.
13. Each customer must pay the bill amount by the specified due date.
14. If the total amount payable exceeds the 3-month billing period amount, then the meter connection should be removed.
15. Maintenance requests must be assigned to the nearest service provider to address them.
16. Each maintenance request must have a description and be associated with Customer\_Id or Supplier\_Id.
17. Each customer can opt for the auto-payment option if the customer has valid credit card details linking to their account.
18. Each customer must have valid account details such as address, phone number, email address, etc.
19. Each usage history record should be associated with a unique Usage\_Id and Customer\_Id.
20. Each usage history record must have a valid date associated with the meter reading of it.
21. Each customer should get a reminder for their bill payment before the due date.
22. Each customer can have multiple maintenance requests.

**Entity-Relationship Diagram**

A diagram of a utility system

Description automatically generated

**Table 1: Components of ERM**

|  |  |  |  |
| --- | --- | --- | --- |
| **Entity** | **Relationship** | **Connectivity** | **Entity** |
| **Billing** | **Customers** | **1:1** | **Customers** |
| **Reminders** | **Bills** | **M:1** | **Billing** |
| **Maintenance\_Requests** | **Maintenance** | **M:1** | **Service\_Provider** |
| **Meter\_Readings** | **Usage** | **1:1** | **Customers** |
| **Auto\_Payments** | **Payments** | **1:1** | **Customers** |
| **Service\_Provider** | **Suppliers** | **1:M** | **Customers** |
| **Region** | **Region** | **1:M** | **Customers** |
| **Usage\_History** | **Usage** | **M:1** | **Customers** |

**Data Dictionary**

The data dictionary is vital when designing a data model as it allows users to quickly access the entities and their requirements. The dataset will suffer from inconsistency if there is no data dictionary. It lists all the entities, including Name, Description, Type, Format, and Range. It also provides the mapping and relationships between the tables, and the connection between them can be traced easily.

**Table 2: Data Dictionary**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [S.no](http://s.no/) | Table Name | Attribute Name | Description | Attribute Type | Format | Range | Required | PK or FK | FK Reference Table |
| 1 | Customers | Customer\_Id | Unique id of the Customer | INT | 9 | 1-999999999 | Y | PK |  |
| Customer\_FName | First name of the customer | VARCHAR(50) | Xxxxxxx |  | Y |  |  |
| Customer\_LName | Last name of the customer | VARCHAR(50) | Xxxxxxx |  | Y |  |  |
| Customer\_Email | Email address of the customer | VARCHAR(50) | Xxxxxxx |  | Y |  |  |
| Customer\_Contact | Contact number of the customer | VARCHAR(10) | Xxxxxxx |  | Y |  |  |
| Customer\_Location | Address of the customer | VARCHAR(150) | Xxxx Xxxxx Xxxxx |  | Y |  |  |
| Supplier\_Id | unique id of the supplier | INT | 9 | 1-999999999 | Y | FK1 | Service\_Provider |
| Region\_Id | Unique id of the region | INT | 9 | 1-999999999 | Y | FK2 | Region |
| Enrolledfor\_Autopayments | Stores boolean type values | BIT | 1/0 |  | Y |  |  |
| 2 | Usage\_History | Usage\_Id | Unique id of Electricity usage | INT | 9 | 1-999999999 | Y | PK |  |
| Customer\_Id | Unique id of the customer | INT | 9 | 1-999999999 | Y | FK1 | Customers |
| Bill\_Generated\_Date | The date on which the electricity bill was generated | DATE | yyyy-mm-dd |  | Y |  |  |
| Bill\_Payment\_Date | The date on which the electricity bill was paid | DATE | yyyy-mm-dd |  | Y |  |  |
| Total\_Amount | The total amount of the bill generated | DECIMAL(999999999,2) | 99999.99 |  | Y |  |  |
| Total\_Consumption | The total consumption of the electricity in bill cycle | DECIMAL(999999999,2) | 99999999.99 |  | Y |  |  |
| 3 | Meter\_Readings | Meter\_Id | Unique id of the meter | INT | 9999 | 1000-999999999 | Y | PK |  |
| Customer\_Id | Unique id of the Customer | INT | 9 | 1-999999999 | Y | FK1 | Customers |
| Meter\_Readings | contains electricity usage | DECIMAL(999999999,2) | 9999999.99 |  | Y |  |  |
| Date | The date which the readings were recorded | DATE | yyyy-mm-dd |  | Y |  |  |
| 4 | Region | Region\_Id | Unique id of the region | INT | 9 | 1-999999999 | Y | PK |  |
| Region\_Name | Name of the region | VARCHAR(50) | Xxxxx |  | Y |  |  |
| 5 | Maintenance\_Requests | Maintenancerequest\_Id | Unique id of the maintenance request | INT | 9 | 1-999999999 | Y | PK |  |
| Supplier\_Id | unique id of the supplier | INT | 9 | 1-999999999 | Y | FK1 | Service\_Provider |
| Region\_Id | Unique id of the region | INT | 9 | 1-999999999 | Y |  |  |
| Customer\_Id | Unique id of the customer | INT | 9 | 1-999999999 | Y |  |  |
| Maintenancerequest\_Date | The date on which the maintenance is requested | DATE | yyyy-mm-dd |  | Y |  |  |
| Description | Description of the maintenance request | VARCHAR(500) | Xxxxxxx Xxxx Xxx |  | N |  |  |
| 6 | Billing | Billing\_Id | Unique Id of the Bill generated | INT | 9 | 1-999999999 | Y | PK |  |
| Billing\_Amount | Total amount of the bill generated based on usage | DECIMAL(999999999,2) | 99999.99 |  | Y |  |  |
| Customer\_Id | Unique id of the Customer | INT | 9 | 1-999999999 | Y | FK1 | Customers |
| Total\_Consumption | The total consumption of the electricity in bill cycle | DECIMAL(999999999,2) | 99999999.99 |  | Y |  |  |
| Supplier\_Id | unique id of the supplier | INT | 9 | 1-999999999 | Y | FK2 | Service\_Provider |
| Billing\_Date | The date on which the bill is generated | DATE | yyyy-mm-dd |  | Y |  |  |
| 7 | Reminders | Reminder\_Id | Unique id of the reminder | INT | 999 | 100-999999999 | Y | PK |  |
| Reminder | Description of the reminder | VARCHAR(250) | Xxxxx Xxxx Xxx |  | Y |  |  |
| Customer\_Id | Unique id of the Customer | INT | 9 | 1-999999999 | Y | FK1 | Customers |
| 8 | Auto\_Payments | Autopayment\_Id | Unique id for auto payment of bill | INT | 9 | 1-999999999 | Y | PK |  |
| Customer\_Id | Unique id of the Customer | INT | 9 | 1-999999999 | Y | FK1 | Customers |
| Card\_Type | Type of the card using | VARCHAR(25) | Xxxxx Xxxxx |  | Y |  |  |
| Card\_Number | Card number of the customer | VARCHAR(19) | XXXX-XXXX-XXXX-XXXX |  | Y |  |  |
| Expiry\_Date | Date of expiry of the card | DATE | yyyy-mm |  | Y |  |  |
| CVV | Three-digit security code for payment | INT | 999 |  | Y |  |  |
| Billing\_Address | Billing address of the customer | VARCHAR(250) | Xxxx Xxxx Xxxx |  | Y |  |  |
| 9 | Service\_Provider | Supplier\_Id | unique id of the supplier | INT | 9 | 1-999999999 | Y | PK |  |
| Supplier\_Name | Name of the Supplier | VARCHAR(50) | Xxxxxx |  | Y |  |  |
| Supplier\_Email | Email id of the Supplier | VARCHAR(50) | Xxxxxxxx |  | Y |  |  |
| Supplieer\_Contact | Contact of the Supplier | VARCHAR(10) | Xxxxxxxxx |  | Y |  |  |

**Queries and Operations**

***Section 1: Entity Generation and Data insertion: Create Database, Create Table and Insert data into the Tables***

**Overview:** EMS data model contains only one database named ‘EMS’ where it lists the database schema for the EMS database and associated tables listed in Table 3. To avoid inserting any type of invalid data, foreign key constraints are applied while the creation of the tables in the database. Each value in the foreign key must have an associated value taken from the primary key.

**(\*Comments:** Before executing queries in section 2, queries in section 1 should be executed avoid errors in extracting the results.)

**Table 3: Database Schema**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Database** | **Table Name** | **Table Explanation** | **Records** | **Insertion order** |
| EMS | Service\_Provider | This table contains information related to the power suppliers |  | 1 |
| EMS | Region | This Table contains all the region names |  | 2 |
| EMS | Customers | This Table provides information about customers |  | 3 |
| EMS | Maintenance\_Requests | This Table contains all the Maintenance requests raised by the customers and suppliers |  | 4 |
| EMS | Meter\_Readings | This Table provides the readings of meter in current billing period |  | 5 |
| EMS | Auto\_Payments | This Table stores the preferred payment method for auto payment of bill in billing period |  | 6 |
| EMS | Billing | This Table provides the total consumption and amount due in current billing period |  | 7 |
| EMS | Reminders | This Table provides the reminders sent to customer related to Billing |  | 8 |
| EMS | Usage\_History | This Table contains all the previous bill payment and consumption history |  | 9 |

**Table 4 Query steps and Explanation**

|  |  |  |
| --- | --- | --- |
| **Steps** | **Query** | **Example** |
| 1.Creating Database | Create database dbname; | Create database EMS; |
| 2.Creating table | Use dbname;  Create table tablename(  A1 datatype,  A2 datatype,  A3 datatype,...  ); | use EMS;  create table Service\_Provider(  Supplier\_Id Int identity(1,1) primary key,  Supplier\_Name Varchar(50) not null,  Supplier\_Email Varchar(50) not null,  Supplier\_Contact Varchar(10) not null  ); |
| 3. Inserting values into table | Use Dbname;  Insert into tablename(A1,A2,A3,..)  Values(B1,B2,B3,…); | use EMS;  insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)  [values('Reliant','8332359653','info@reliant.com](mailto:values('Reliant','8332359653','info@reliant.com)'); |
| 4.Retrieving the data | Use dbname;  Select \* from tablename; | use EMS;  select \* from Service\_Provider; |

**Entity Generation and Data Entry for Table Service\_Provider:**

**Statements Explanation**

* The Database EMS is created in the database schema using command “Create database EMS”.
* The Table Service\_Provider is created in EMS database using command “Create table Service\_Provider” and command “Use EMS” is used before creating the table to store the table in EMS database.
* Table Service\_Provider is filled with relevant data using command “Insert into” and inserting values into the table.
* Finally, the result will be displayed by using command “select \* from Service\_Provider” to display all inserted values in the table.

**Queries**

/\*------------------------------------------Create database EMS--------------------------------------------\*/

Create database EMS;

/\*--------------------------------------Create Service\_Provider Table-------------------------------------\*/

use EMS;

create table Service\_Provider(

Supplier\_Id Int identity(1,1) primary key,

Supplier\_Name Varchar(50) not null,

Supplier\_Email Varchar(50) not null,

Supplier\_Contact Varchar(10) not null

);

/\*------------------------------------Insert values to Service\_Provider Table---------------------------\*/

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Reliant','8332359653','info@reliant.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Oncor','8263758293','info@oncor.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Compare Power','8462758364','info@comparepowwer.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Amigo Power','9162553475','info@Amigopower.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Ampra Energy','7253856495','info@amprapower.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Energy Texas','8371698365','info@energytexas.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Tara energy','7239476849','info@taraenergy.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Onver Power','5837469382','info@onverpower.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Tata Powers','8385945848','info@tatapower.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Cupa power','5837485737','info@cupapower.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Zampa Distributions','4938572647','info@zampadistributions.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('spectrum powers','9472837485','info@tspectrumpowers.com');

/\*--------------------------------------------------------------------------------------------------------------\*/

**Result**

Table Explanation: Service\_provider provides information about suppliers like their name, contact, and email. To query all the information command select \* from service\_Provider” is used.

***Query***

select \* from Service\_Provider;

***Screenshot***

A screenshot of a computer

Description automatically generated

**Entity Generation and Data Entry for Table Region:**

**Statements Explanation**

* The database EMS is already created. Command “Use EMS” is used to call the database.
* Table Region is created in EMS database. Command “Create table Region” is used to create the table.
* The table Region is filled with relevant data using command “Insert into \*\*”.
* Then, the results will be displayed by using the command “select \* from Region” to query all the inserted values to the table.

**Queries**

/\*--------------------------------------------Create table Region--------------------------------------------\*/

create table Region(

Region\_Id int identity(1,1) primary key,

Region\_Name varchar(50) not null

);

/\*---------------------------------------Inserting values into Region table--------------------------------\*/

insert into Region values('Denton');

insert into Region values('Dallas');

insert into Region values('Houston');

insert into Region values('Austin');

insert into Region values('Irving');

insert into Region values('Plano');

insert into Region values('Frisco');

insert into Region values('Fort-worth');

insert into Region values('Lewisville');

insert into Region values('Phoenix');

/\*----------------------------------------------------------------------------------------------------------------\*/

**Result**

Table Explanation: Region table provides information about the region ID and name.

***Query***

select \* from Region;

***Screenshot***

A screenshot of a computer

Description automatically generated

**Entity Generation and Data Entry for Table Customers:**

**Statements Explanation**

* The database EMS is already created. Command “Use EMS” is used to call the database.
* Table Customers is created in EMS database. Command “Create table Customers” is used to create the table.
* The table Customers is filled with relevant data using command “Insert into \*\*”.
* Then, the results will be displayed by using the command “select \* from Customers” to query all the inserted values to the table.

**Queries**

/\*----------------------------------------------Create table Customers-------------------------------------\*/

create table Customers(

Customer\_Id int identity(1,1) primary key,

Customer\_FName varchar(50) not null,

Customer\_LName varchar(50) not null,

Customer\_Email varchar(50) not null,

Customer\_Contact varchar(10) not null,

Customer\_Location varchar(150) not null,

Supplier\_Id int not null,

Region\_Id int not null,

Enrolledfor\_Autopayments bit default 0 not null,

foreign key(Supplier\_Id) references Service\_Provider(Supplier\_Id),

foreign key(Region\_Id) references Region(Region\_Id)

);

/\*----------------------------------------Inserting values into Customers table---------------------------\*/

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Raghu','nandhan','4857362847','raghu@gmail.com','union circle, denton, tx',1,1,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('ram','charan','4837285674','ram@gmail.com','oaks of denton, denton, TX',3,1,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Email,Customer\_Contact,Customer\_Location,Supplier\_Id,Region\_Id)

values('Rishi','kumar','rishi@gmail.com','8573746385','valley ranch pkwy, Irving, TX',1,5);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('priya','chandran','9472847364','pricha@gmail.com','oaks at valley ranch, Irving, Tx',2,5,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Sruthi','Kiran','9263828462','kiran@gmail.com','Swadeshi plaza, Irving, TX',4,5,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Cristina','Rose','6284638472','cristina@gmail.com','Wilmington pkwy, Fort-worth, Tx',4,8,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Ronald','Wright','7483648264','ronald@gmail.com','Dallas pkwy, dallas, Tx',6,2,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Terissa','Tim','9273846284','teressa@gmail.com','Loscolinas pkwy, houston, Tx',5,3,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Mike','Tong','9374738472','Mike@gmail.com','Spring valley rd,plano, Tx',3,6,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Tom','Crissey','8284637463','Tom@gmail.com','Spring valley rd,plano, Tx',2,6,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Micheal','Tom','8264927482','Micheal@gmail.com','Montfort dr, frisco, Tx',1,7,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Pavan','Naini','9405959573','pavan@gmail.com','Ioof St, Houston, Tx',9,3,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Kalyan','Tinku','9890987890','tinku@gmail.com','Bernald St denton, Tx',1,1,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Kiran Kumar','Mintu','9309439890','kirankumar@gmail.com','lebin street, lw',7,9,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Adam','Smith','9453245679','adamsmith@gmail.com','Teasley Lane,Ph',7,10,12);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Anvar','mith','8293746738','anvar@gmail.com','Timber st, Lewisville, Tx',11,9,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('cane','walter','3947584839','cane@gmail.com','Twistley rd,Frisco,Tx',10,7,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Kane','county','5928385858','kane@gmail.com','winter ln, Plano, Tx',8,6,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Ruth','ranger','7838485868','ruth@gmail.com','Rinder pkwy, Dallas, Tx',6,2,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Birkey','wing','8273648573','birkey@gmail.com','oaks of denton, Denton, Tx',11,1,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('king','william','7394829485','kingwilliam@gmail.com','trails of austin, Austin,Tx',12,4,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('ram','singh','8483847275','ramsingh@gmail.com','lukes of fort-worth, Fort-worth, Tx',9,8,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Trint','bing','8938475837','trinth@gmail.com','churchy st, Phoenix, Tx',5,10,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('wright','wrong','8394388484','wright@gmail.com','trinty ln, Denton, Tx',8,1,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('miller','wright','8273645273','miller@gmail.com','west street ln, Irving, Tx',9,5,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Anil','kumbley','7483746574','kumbley@gmail.com','Timber west ln, Austin, Tx',2,4,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('virat','rona','7384958274','virat@gmail.com','valley pkwy E, Plano, Tx',4,6,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('vikranth','rana','8938485828','vikranth@gmail.com','ruster rd, Fort-worth, Tx',8,8,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Ringer','rale','8394727475','ringer@gmail.com','winster pkwy, Austin, Tx',10,4,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('stella','fring','9284837475','stella@gmail.com','oaks of Phoenix, Phoenix, Tx',11,10,0);

/\*----------------------------------------------------------------------------------------------------------------\*/

**Result**

Table Explanation: Table Customers provides information about all customer information like name, contact, email and address.

***Query***

select \* from Customers;

***Screenshot***

A screenshot of a computer

Description automatically generated

**Entity Generation and Data Entry for Table Maintenance\_Requests:**

**Statements Explanation**

* The database EMS is already created. Command “Use EMS” is used to call the database.
* Table Maintenance\_Requests is created in EMS database. Command “Create table Maintenance\_Requests” is used to create the table.
* The table Maintenance\_Requests is filled with relevant data using command “Insert into \*\*”.
* Then, the results will be displayed by using the command “select \* from Maintenance\_Requests” to query all the inserted values to the table.

**Queries**

/\*------------------------------------------Create Maintenance\_Requests Table--------------------------\*/

create table Maintenance\_Requests(

Maintenancerequest\_Id int identity(1,1) primary key,

Supplier\_Id int not null,

Region\_Id int default null,

Customer\_Id int default null,

Maintenancerequest\_Date date not null,

Description varchar(500) default' ',

foreign key(Supplier\_Id) REFERENCES Service\_Provider(Supplier\_Id)

);

/\*--------------------------------Inserting values into Maintenance\_Requests Table-------------------\*/

insert into Maintenance\_Requests(Supplier\_Id,Maintenancerequest\_Date,[Description])

values(2,'2023-06-22','Maintenance is scheduled at the supplier end annually');

insert into Maintenance\_Requests

values(1,5,4,'2023-08-23','Request for Installation of meter');

insert into Maintenance\_Requests

values(1,1,1,'2023-06-12','Request for Installation of meter');

insert into Maintenance\_Requests(Supplier\_Id,Maintenancerequest\_Date,[Description])

values(5,'2023-08-11','Maintenance is scheduled at the supplier end annually');

insert into Maintenance\_Requests

values(4,8,6,'2023-04-23','Request for Installation of meter');

insert into Maintenance\_Requests

values(6,2,7,'2023-10-05','Request for Installation of meter');

insert into Maintenance\_Requests

values(5,3,8,'2023-09-02','Request for Installation of meter');

insert into Maintenance\_Requests

values(2,6,10,'2023-08-06','Request for Installation of meter');

insert into Maintenance\_Requests

values(1,7,11,'2023-10-01','Request for Installation of meter');

insert into Maintenance\_Requests

values(1,1,13,'2023-12-02','Request for Installation of meter');

insert into Maintenance\_Requests

values(7,10,15,'2023-11-04','Request for change of meter');

insert into Maintenance\_Requests

values(10,7,17,'2023-08-13','Request for Installation of meter');

insert into Maintenance\_Requests(Supplier\_Id,Maintenancerequest\_Date,[Description])

values(3,'2023-11-10','Maintenance is scheduled at the supplier end annually');

insert into Maintenance\_Requests

values(6,12,19,'2023-09-12','Request for change of meter');

insert into Maintenance\_Requests

values(11,1,20,'2023-08-13','Request for Installation of meter');

insert into Maintenance\_Requests

values(9,8,22,'2023-10-04','Request for change of meter');

insert into Maintenance\_Requests

values(5,10,23,'2023-11-13','Request for Installation of meter');

insert into Maintenance\_Requests

values(2,4,26,'2023-09-06','Request for change of meter');

insert into Maintenance\_Requests

values(8,8,28,'2023-08-03','Request for Installation of meter');

insert into Maintenance\_Requests

values(10,4,29,'2023-07-12','Request for Installation of meter');

/\*----------------------------------------------------------------------------------------------------------------\*/

**Result**

Table Explanation: Maintenance\_Requests table provides the information about the maintenance requests raised by the customers and suppliers.

***Query***

select \* from Maintenance\_Requests;

***Screenshot***

A screenshot of a computer

Description automatically generated

**Entity Generation and Data Entry for Table Meter\_Readings:**

**Statements Explanation**

* The database EMS is already created. Command “Use EMS” is used to call the database.
* Table Meter\_Readings is created in EMS database. Command “Create table Meter\_Readings” is used to create the table.
* The table Meter\_Readings is filled with relevant data using command “Insert into \*\*”.
* Then, the results will be displayed by using the command “select \* from Meter\_Readings” to query all the inserted values to the table.

**Queries**

/\*---------------------------------------------Create Meter\_Readings Table-------------------------------\*/

create table Meter\_Readings(

Meter\_Id int identity(1000,1) primary key,

Customer\_Id int not null,

Meter\_Readings decimal(9,2) default 0.00,

Readings\_Date date not null,

foreign key(Customer\_Id) references Customers(Customer\_Id)

);

/\*----------------------------------Inserting values into Meter\_Readings Table-------------------------\*/

insert into Meter\_Readings

values(1,23.56,'2023-11-20');

insert into Meter\_Readings

values(2,13.27,'2023-11-20');

insert into Meter\_Readings

values(3,20.46,'2023-11-20');

insert into Meter\_Readings

values(4,32.76,'2023-11-20');

insert into Meter\_Readings

values(5,53.2,'2023-11-20');

insert into Meter\_Readings

values(6,49.5,'2023-11-20');

insert into Meter\_Readings

values(7,42.3,'2023-11-20');

insert into Meter\_Readings

values(8,64.2,'2023-11-20');

insert into Meter\_Readings

values(9,40.3,'2023-11-20');

insert into Meter\_Readings

values(10,54.5,'2023-11-20');

insert into Meter\_Readings

values(11,46.7,'2023-11-20');

insert into Meter\_Readings

values(12,34.4,'2023-11-20');

insert into Meter\_Readings

values(13,10.1,'2023-11-20');

insert into Meter\_Readings

values(14,19.2,'2023-11-20');

insert into Meter\_Readings

values(15,43.2,'2023-11-20');

insert into Meter\_Readings

values(16,53.5,'2023-11-20');

insert into Meter\_Readings

values(17,23.27,'2023-11-20');

insert into Meter\_Readings

values(18,22.46,'2023-11-20');

insert into Meter\_Readings

values(19,22.76,'2023-11-20');

insert into Meter\_Readings

values(20,12.51,'2023-11-20');

insert into Meter\_Readings

values(21,33.56,'2023-11-20');

insert into Meter\_Readings

values(22,23.27,'2023-11-20');

insert into Meter\_Readings

values(23,25.46,'2023-11-20');

insert into Meter\_Readings

values(24,22.76,'2023-11-20');

insert into Meter\_Readings

values(25,16.51,'2023-11-20');

insert into Meter\_Readings

values(26,31.76,'2023-11-20');

insert into Meter\_Readings

values(27,19.27,'2023-11-20');

insert into Meter\_Readings

values(28,27.46,'2023-11-20');

insert into Meter\_Readings

values(29,37.76,'2023-11-20');

insert into Meter\_Readings

values(30,19.51,'2023-11-20');

/\*----------------------------------------------------------------------------------------------------------------\*/

**Result**

Table Explanation: Meter\_Readings table provides the information about the electricity usage by the customer with specific meter\_Id in current billing period.

***Query***

select \* from Meter\_Readings;

***Screenshot***

A screenshot of a computer

Description automatically generated

**Entity Generation and Data Entry for Table Auto\_Payments:**

**Statements Explanation**

* The database EMS is already created. Command “Use EMS” is used to call the database.
* Table Auto\_Payments is created in EMS database. Command “Create table Auto\_Payments” is used to create the table.
* The table Auto\_Payments is filled with relevant data using command “Insert into \*\*”.
* Then, the results will be displayed by using the command “select \* from Auto\_Payments” to query all the inserted values to the table.

**Queries**

/\*---------------------------------------------Create Auto\_Payments Table-------------------------------\*/

create table Auto\_Payments(

Autopayment\_Id int identity(1,1) primary key,

Customer\_Id int unique not null,

Card\_Type varchar(25) not null,

Card\_Number varchar(19) not null unique,

Expiry\_Date Date not null,

CVV int not null,

Billing\_Address varchar(250) not null,

foreign key(Customer\_Id) REFERENCES Customers(Customer\_Id)

);

/\*----------------------------------Inserting values into Auto\_Payments Table-------------------------\*/

insert into Auto\_Payments

values(1,'credit','4783927594716258','2024-02-01',346,'union circle');

insert into Auto\_Payments

values(4,'debit','8264857102859374','2025-12-01',153,'oaks at valley ranch');

insert into Auto\_Payments

values(6,'Debit','5263846273849270','2025-12-01',134,'Wilmington pkwy, Fort-worth, tx');

insert into Auto\_Payments

values(8,'debit','8263745261846370','2026-04-01',143,'Loscolinas pkwy, houston, tx');

insert into Auto\_Payments

values(11,'debit','8394728463746370','2024-03-01',256,'Montfort dr, frisco, tx');

insert into Auto\_Payments

values(12,'Credit','12345','2027-01-01',123,'Ioof st,tx');

insert into Auto\_Payments

values(14,'Debit','2341234564332','2030-11-01',533,'lebin street, lw');

insert into Auto\_Payments

values(15,'Credit','234567812234','2025-12-01',324,'bernard street,denton');

insert into Auto\_Payments

values(16,'credit','2150816150957480','2031-01-01',587,'Discovery park dr');

insert into Auto\_Payments

values(17,'credit','4838485969485860','2024-07-01',564,'Twistley rd,Frisco,Tx');

insert into Auto\_Payments

values(18,'debit','9483847562736450','2025-06-01',498,'winter ln, Plano, Tx');

insert into Auto\_Payments

values(23,'credit','2938485862949280','2024-08-01',734,'churchy st, Phoenix, Tx');

insert into Auto\_Payments

values(27,'debit','7374859384759280','2027-11-01',435,'valley pkwy E, Plano, Tx');

/\*----------------------------------------------------------------------------------------------------------------\*/

**Result**

Table Explanation: Auto\_Payments table provides the information about the preferred payment method by the customer to automatically debit the billing mount from the card details provided by the customers.

***Query***

select \* from Auto\_Payments;

***Screenshot***

A screenshot of a computer

Description automatically generated

**Entity Generation and Data Entry for Table Billing:**

**Statements Explanation**

* The database EMS is already created. Command “Use EMS” is used to call the database.
* Billing table is created in EMS database. Command “Create table Billing” is used to create the table.
* The table Billing is filled with relevant data using command “Insert into \*\*”.
* Then, the results will be displayed by using the command “select \* from Billing” to query all the inserted values to the table.

**Queries**

/\*---------------------------------------------Create Billing Table-------------------------------\*/

create table Billing(

Billing\_Id int identity(1,1) primary key,

Billing\_Amount decimal(9,2) default 0.00,

Customer\_Id int unique not null,

Total\_Consumption decimal(9,2) default 0.00,

Supplier\_Id int not null,

Billing\_Date date,

foreign key(Customer\_Id) references Customers(Customer\_Id)

);

/\*----------------------------------Inserting values into Billing Table-------------------------\*/

insert into Billing

values(56.45,1,134.45,1,'2023-11-03');

insert into Billing

values(43.25,2,108.42,2,'2023-11-03');

insert into Billing

values(72.34,3,184.3,1,'2023-11-03');

insert into Billing

values(38.34,4,90.2,2,'2023-11-03');

insert into Billing

values(58.45,5,105.32,4,'2023-11-03');

insert into Billing

values(52.2,6,105.36,4,'2023-11-03');

insert into Billing

values(48.4,7,89.9,6,'2023-11-03');

insert into Billing

values(72.2,8,130.36,5,'2023-11-03');

insert into Billing

values(44.43,9,82.6,3,'2023-11-03');

insert into Billing

values(54.9,10,103.5,2,'2023-11-03');

insert into Billing

values(48.89,11,86.76,1,'2023-11-03');

insert into Billing

values(50.55,12,75,9,'2023-11-03');

insert into Billing

values(12,13,31,1,'2023-11-03');

insert into Billing

values(45,14,12,7,'2023-11-03');

insert into Billing

values(94,15,56,7,'2023-11-03');

insert into Billing

values(100,16,200,11,'2023-11-03');

insert into Billing

values(86.65,17,164.6,10,'2023-11-03');

insert into Billing

values(46.87,18,128.8,8,'2023-11-03');

insert into Billing

values(55.23,19,104.45,6,'2023-11-03');

insert into Billing

values(37.87,20,74.45,11,'2023-11-03');

insert into Billing

values(64.57,21,124.45,12,'2023-11-03');

insert into Billing

values(37.43,22,84.55,9,'2023-11-03');

insert into Billing

values(42.45,23,90.43,5,'2023-11-03');

insert into Billing

values(39.78,24,80.15,8,'2023-11-03');

insert into Billing

values(37.67,25,82.55,9,'2023-11-03');

insert into Billing

values(46.23,26,101.86,2,'2023-11-03');

insert into Billing

values(48.65,27,96.23,4,'2023-11-03');

insert into Billing

values(39.78,28,92.45,8,'2023-11-03');

insert into Billing

values(52.78,29,108.36,10,'2023-11-03');

insert into Billing

values(49.67,30,103.28,11,'2023-11-03');

/\*----------------------------------------------------------------------------------------------------------------\*/

**Result**

Table Explanation: Billing table provides the information about customer’s total consumption in the previous bill cycle and amount due.

***Query***

select \* from Billing;

***Screenshot***

A screenshot of a computer

Description automatically generated

**Entity Generation and Data Entry for Table Reminders:**

**Statements Explanation**

* The database EMS is already created. Command “Use EMS” is used to call the database.
* Reminders table is created in EMS database. Command “Create table Reminders” is used to create the table.
* The table Reminders is filled with relevant data using command “Insert into \*\*”.
* Then, the results will be displayed by using the command “select \* from Reminders” to query all the inserted values to the table.

**Queries**

/\*--------------------------------------------- Reminders Table-------------------------------\*/

create table Reminders(

Reminder\_Id int identity(100,1) primary key,

Reminder varchar(250) default 'No reminders or changed reminder preferences',

Customer\_Id int not null,

foreign key(Customer\_Id) REFERENCES Customers(Customer\_Id)

);

/\*----------------------------------Inserting values into Reminders Table-------------------------\*/

insert into Reminders

VALUES('Your bill has paid usig autopayment method',1);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',2);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',3);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',4);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',5);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',6);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',7);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',8);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',9);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',10);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',11);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',12);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',13);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',14);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',15);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',16);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',17);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',18);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',19);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',20);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',21);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',22);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',23);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',24);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',25);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',26);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',27);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',28);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',29);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',30);

/\*----------------------------------------------------------------------------------------------------------------\*/

**Result**

Table Explanation: Reminders table provides the information about the reminders sent to customers regarding the bill payments.

***Query***

select \* from Reminders;

***Screenshot***

A screenshot of a computer

Description automatically generated

**Entity Generation and Data Entry for Table Usage\_History:**

**Statements Explanation**

* The database EMS is already created. Command “Use EMS” is used to call the database.
* Usage\_History table is created in EMS database. Command “Create table Usage\_History” is used to create the table.
* The table Usage\_History is filled with relevant data using command “Insert into \*\*”.
* Then, the results will be displayed by using the command “select \* from Usage\_History” to query all the inserted values to the table.

**Queries**

/\*----------------------------------------------Usage\_History Table-------------------------------\*/

create table Usage\_History(

Usage\_Id int identity(1,1) primary key,

Customer\_Id int not null,

Bill\_Generated\_Date date not null,

Bill\_payment\_Date date not null,

Total\_Amount decimal(9,2),

Total\_Consumption decimal(9,2),

foreign key(Customer\_Id) references Customers(Customer\_Id)

);

/\*----------------------------------Inserting values into Usage\_History Table-------------------------\*/

insert into Usage\_History

values(1,'2023-10-3','2023-10-18',56.78,126.54);

insert into Usage\_History

values(1,'2023-09-3','2023-09-15',53.18,106.54);

insert into Usage\_History

values(1,'11/3/2023','11/6/2023',46.77,92.5);

insert into Usage\_History

values(2,'2023-10-3','2023-10-18',46.78,88.54);

insert into Usage\_History

values(2,'11/3/2023','11/12/2023',36.77,72.5);

insert into Usage\_History

values(3,'11/3/2023','11/11/2023',33.7,66.9);

insert into Usage\_History

values(4,'11/3/2023','11/6/2023',20.3,40.6);

insert into Usage\_History

values(5,'2023-10-3','2023-10-18',36.3,70.1);

insert into Usage\_History

values(5,'11/3/2023','11/07/2023',36.5,70.2);

insert into Usage\_History

values(6,'11/3/2023','11/6/2023',33.6,67.3);

insert into Usage\_History

values(7,'11/3/2023','11/08/2023',29.6,60.4);

insert into Usage\_History

values(8,'11/3/2023','11/6/2023',39.5,80.4);

insert into Usage\_History

values(9,'11/3/2023','11/15/2023',60.3,130.4);

insert into Usage\_History

values(10,'11/3/2023','11/8/2023',53.5,104.3);

insert into Usage\_History

values(11,'2023-10-3','2023-06-18',44.7,95.6);

insert into Usage\_History

values(11,'11/3/2023','11/6/2023',50.4,100.8);

insert into Usage\_History

values(12,'2023-10-3','2023-06-18',39.5,80.9);

insert into Usage\_History

values(12,'2023-09-3','2023-06-15',37.8,75.8);

insert into Usage\_History

values(12,'11/3/2023','11/6/2023',41.7,90.2);

insert into Usage\_History

values(13,'11/3/2023','11/13/2023',33.33,66.66);

insert into Usage\_History

values(14,'11/3/2023','11/6/2023',42.27,82.7);

insert into Usage\_History

values(15,'2023-10-3','2023-06-18',52.5,102.54);

insert into Usage\_History

values(15,'11/3/2023','11/6/2023',37.77,82.54);

insert into Usage\_History

values(16,'11/3/2023','11/6/2023',66.77,122.5);

insert into Usage\_History

values(17,'2023-10-3','2023-06-18',37.5,60.5);

insert into Usage\_History

values(17,'2023-09-3','2023-06-15',35.9,69.9);

insert into Usage\_History

values(17,'11/3/2023','11/6/2023',35.7,69.8);

insert into Usage\_History

values(18,'11/3/2023','11/08/2023',37.9,77.8);

insert into Usage\_History

values(19,'2023-10-3','2023-07-18',41.5,92.4);

insert into Usage\_History

values(19,'11/3/2023','11/05/2023',34.9,74.3);

insert into Usage\_History

values(20,'11/3/2023','11/10/2023',44.4,88.8);

insert into Usage\_History

values(21,'11/3/2023','11/15/2023',28.6,52.4);

insert into Usage\_History

values(22,'11/3/2023','11/18/2023',34.8,74.6);

insert into Usage\_History

values(23,'2023-10-3','2023-06-18',39.6,80.3);

insert into Usage\_History

values(23,'2023-09-3','2023-06-15',36.7,71.4);

insert into Usage\_History

values(23,'11/3/2023','11/06/2023',38.9,72.4);

insert into Usage\_History

values(24,'2023-10-3','2023-12-18',47.4,90.3);

insert into Usage\_History

values(24,'11/3/2023','11/6/2023',55.9,112.5);

insert into Usage\_History

values(25,'2023-10-3','2023-10-18',44.8,90.5);

insert into Usage\_History

values(25,'2023-09-3','2023-09-15',51.7,103.2);

insert into Usage\_History

values(25,'11/3/2023','11/12/2023',41.7,83.4);

insert into Usage\_History

values(26,'2023-10-3','2023-11-18',37.6,72.4);

insert into Usage\_History

values(26,'11/3/2023','11/13/2023',33.6,66.7);

insert into Usage\_History

values(27,'2023-10-3','2023-06-18',61.2,120.5);

insert into Usage\_History

values(27,'2023-09-3','2023-06-15',41.4,83.2);

insert into Usage\_History

values(27,'11/3/2023','11/06/2023',39.5,76.8);

insert into Usage\_History

values(28,'2023-10-3','2023-08-18',53.4,101.2);

insert into Usage\_History

values(28,'11/3/2023','11/12/2023',28.9,60.2);

insert into Usage\_History

values(29,'2023-10-3','2023-09-18',36.7,72.2);

insert into Usage\_History

values(29,'11/3/2023','11/15/2023',54.6,102.2);

insert into Usage\_History

values(30,'2023-10-3','2023-05-18',45.38,90.5);

insert into Usage\_History

values(30,'11/3/2023','11/06/2023',43.34,86.9);

/\*----------------------------------------------------------------------------------------------------------------\*/

**Result**

Table Explanation: Usage\_History table provides the information about the reminders sent to customers regarding the bill payments.

***Query***

select \* from Usage\_History;

***Screenshot***

A screenshot of a computer

Description automatically generated

***Section 2: Data Retrieval and Analytical Reports***

**Overview:** Raw data can be used to extract useful information using SQL, and it plays a major role in the data retrieval. This retrieved data can further be used by any company to generate insights such as generating analytics or reports, and to make data driven decisions. Here in this section of data analysis, SQL queries are generated using the SQL SELECT statement along with the brief explanation of the statement, and the result displayed is captured as a screenshot. **(\*Comments:** Before executing queries in this section 2, queries in section 1 should be executed avoid errors in extracting the results.)

**Data Analysis 1**

***Statement:*** Get all the details related to customer and sort it by the customer first name.

Sql command explanation:

* Command select \* is used to get all the columns from the table.
* Command from is used to retrive data from the table.
* Command order by Customer\_FName is used to sort the result by the Customer

***SQL query:***

select \* from Customers

order by Customer\_FName;

***Result and screenshot:***

A screenshot of a computer

Description automatically generated

**Data Analysis 2**

***Statement:*** Get all the details related to supplier and sort it by the supplier name.

Sql command explanation:

* Command select \* is used to get all the columns from the table.
* Command from is used to retrive data from the table.
* Command order by supplier name is used to sort the result by the supplier name.

***SQL query:***

select \* from Service\_Provider

order by Supplier\_Name;

***Result and screenshot:***

A screenshot of a computer

Description automatically generated

**Data Analysis 3**

***Statement:*** Get the total number of customers in a region and sort it by region.

Sql command explanation:

* Command “Use EMS” is used to call the EMS database.
* Command count(\*) is used to get the number of customers.
* Command as is used to rename the column count(\*) as an alias.
* Command join is used to join two tables together with ON to specify which column is used to join the table.
* Command group by Region is used to aggregate the data by Region.
* Command order by Region is used to sort the result by the Region name.

***SQL query:***

select region\_name, count(\*) as No\_Customers from customers c

join region r on c.Region\_Id=r.Region\_Id

group by region\_name

order by region\_name;

***Result and screenshot:***

A screenshot of a computer

Description automatically generated

**Data Analysis 4**

***Statement:*** Get the total number of customers who signed-up for supplier and sort it by supplier name.

Sql command explanation:

* Command count(\*) is used to get the number of customers.
* Command as is used to rename the column count(\*) as an alias.
* Command join is used to join two tables together with ON to specify which column is used to join the table.
* Command group by Supplier\_name is used to aggregate the data by Supplier\_Name.
* Command order by Supplier\_name is used to sort the result by the Supplier\_Name.

***SQL query:***

select Supplier\_name, count(\*) as No\_Customers from customers c

join Service\_Provider s on c.supplier\_Id=s.supplier\_Id

group by supplier\_name

order by supplier\_name;

***Result and screenshot:***

A screenshot of a computer

Description automatically generated

**Data Analysis 5**

***Statement:*** List the customer name, reminders sent to customers and sort it by customer name.

Sql command explanation:

* Command from is used to retrive data from the table.
* Command join is used to join two tables together with ON to specify which column is used to join the table.
* Command order by Customer\_name is used to sort the result by the Customer\_Name.

***SQL query:***

select customer\_Fname,Customer\_LName, r.Reminder from Customers c

join Reminders r on c.Customer\_Id=r.Customer\_Id

order by Customer\_FName;

***Result and screenshot:***

A screenshot of a computer

Description automatically generated

**Data Analysis 6**

***Statement:*** Get the meter readings and the customer full name and sort it by the meter readings.

Sql command explanation:

* Command from is used to retrive data from the table.
* Command join is used to join two tables together with ON to specify which column is used to join the table.
* Command order by Meter\_Readings is used to sort the result by the Meter\_Readings

***SQL query:***

select customer\_Fname,Customer\_LName, Meter\_Readings from Customers c

join Meter\_Readings m on c.customer\_Id=m.Customer\_Id

order by Meter\_Readings;

***Result and screenshot:***

A screenshot of a computer

Description automatically generated

**Data Analysis 7**

***Statement:*** Get the billing details of customer whose billing amount is over $50 and order the result in descending order using bill amount.

Sql command explanation:

* Command from is used to retrive data from the table.
* Command join is used to join two tables together with ON to specify which column is used to join the table.
* Command where is used to apply the condition to the query where the bill amount is over 50$.
* Command order by Billing\_Amount is used to sort the result by the Billing\_Amount.

***SQL query:***

select Customer\_FName,Customer\_LName,Billing\_Amount from Customers c

join Billing b on c.Customer\_Id=b.Customer\_Id

where Billing\_Amount>50;

***Result and screenshot:***

A screenshot of a computer

Description automatically generated

**Data Analysis 8**

***Statement:*** Get the maintenance requests description raised by the customers with date and sort it by the request date.

Sql command explanation:

* Command from is used to retrive data from the table.
* Command join is used to join two tables together with ON to specify which column is used to join the table.
* Command order by Maintenancerequest\_Date is used to sort the result by the Maintenancerequest\_Date.

***SQL query:***

select Customer\_FName,Customer\_LName,[Description],Maintenancerequest\_Date from Customers c

join Maintenance\_Requests m on c.Customer\_Id=m.Customer\_Id

order by Maintenancerequest\_Date;

***Result and screenshot:***

A screenshot of a computer

Description automatically generated

**Data Analysis 9**

***Statement:*** Get the usage history of the customers in the month of November whose billing amount is over 30$ and sort it by bill payment date.

Sql command explanation:

* Command from is used to retrive data from the table.
* Command join is used to join two tables together with ON to specify which column is used to join the table.
* Command where is used to apply the condition to the query where the bill amount is over 30$ and bill generated in the month November.
* Command order by Bill\_Payment\_Date is used to sort the result by the Bill\_Payment\_Date.

***SQL query:***

select Customer\_FName,Customer\_LName ,Bill\_Generated\_Date,Bill\_payment\_Date,Total\_Amount,Total\_Consumption from Usage\_History u

join Customers c on u.Customer\_Id=c.Customer\_Id

where MONTH(Bill\_Generated\_Date)=11 and Total\_Amount>30

order by Bill\_payment\_Date;

***Result and screenshot:***

A screenshot of a computer

Description automatically generated

**Data Analysis 10**

***Statement:*** Get the total consumption of electricity from suppliers which is used by the customers and display the column as Total\_Consumption.

Sql command explanation:

* Command from is used to retrive data from the table.
* Command join is used to join two or more tables together with ON to specify which column is used to join the table.
* Command group by Supplier\_name is used to aggregate the data by Supplier\_Name.
* Command order by total\_consumption is used to sort the result by the total\_Consumption.

***SQL query:***

select supplier\_name, sum(Meter\_Readings) as Total\_Consumption from Service\_Provider s

join Customers c on s.Supplier\_Id=c.Supplier\_Id

join Meter\_Readings m on c.Customer\_Id=m.Customer\_Id

group by Supplier\_Name;

***Result and screenshot:***

A screenshot of a computer

Description automatically generated

**Appendix 1**

/\*------------------------------------------Create database EMS--------------------------------------------\*/

create database EMS;

/\*--------------------------------------Create Service\_Provider Table-------------------------------------\*/

use EMS;

create table Service\_Provider(

Supplier\_Id Int identity(1,1) primary key,

Supplier\_Name Varchar(50) not null,

Supplier\_Email Varchar(50) not null,

Supplier\_Contact Varchar(10) not null

);

/\*--------------------------------------------Create table Region--------------------------------------------\*/

create table Region(

Region\_Id int identity(1,1) primary key,

Region\_Name varchar(50) not null

);

/\*----------------------------------------------Create table Customers-------------------------------------\*/

create table Customers(

Customer\_Id int identity(1,1) primary key,

Customer\_FName varchar(50) not null,

Customer\_LName varchar(50) not null,

Customer\_Email varchar(50) not null,

Customer\_Contact varchar(10) not null,

Customer\_Location varchar(150) not null,

Supplier\_Id int not null,

Region\_Id int not null,

Enrolledfor\_Autopayments bit default 0 not null,

foreign key(Supplier\_Id) references Service\_Provider(Supplier\_Id),

foreign key(Region\_Id) references Region(Region\_Id)

);

/\*------------------------------------------Create Maintenance\_Requests Table--------------------------\*/

create table Maintenance\_Requests(

Maintenancerequest\_Id int identity(1,1) primary key,

Supplier\_Id int not null,

Region\_Id int default null,

Customer\_Id int default null,

Maintenancerequest\_Date date not null,

Description varchar(500) default' ',

foreign key(Supplier\_Id) REFERENCES Service\_Provider(Supplier\_Id)

);

/\*---------------------------------------------Create Meter\_Readings Table-------------------------------\*/

create table Meter\_Readings(

Meter\_Id int identity(1000,1) primary key,

Customer\_Id int not null,

Meter\_Readings decimal(9,2) default 0.00,

Readings\_Date date not null,

foreign key(Customer\_Id) references Customers(Customer\_Id)

);

/\*---------------------------------------------Create Auto\_Payments Table-------------------------------\*/

create table Auto\_Payments(

Autopayment\_Id int identity(1,1) primary key,

Customer\_Id int unique not null,

Card\_Type varchar(25) not null,

Card\_Number varchar(19) not null unique,

Expiry\_Date Date not null,

CVV int not null,

Billing\_Address varchar(250) not null,

foreign key(Customer\_Id) REFERENCES Customers(Customer\_Id)

);

/\*---------------------------------------------Create Billing Table-------------------------------\*/

create table Billing(

Billing\_Id int identity(1,1) primary key,

Billing\_Amount decimal(9,2) default 0.00,

Customer\_Id int unique not null,

Total\_Consumption decimal(9,2) default 0.00,

Supplier\_Id int not null,

Billing\_Date date,

foreign key(Customer\_Id) references Customers(Customer\_Id)

);

/\*--------------------------------------------- Reminders Table-------------------------------\*/

create table Reminders(

Reminder\_Id int identity(100,1) primary key,

Reminder varchar(250) default 'No reminders or changed reminder preferences',

Customer\_Id int not null,

foreign key(Customer\_Id) REFERENCES Customers(Customer\_Id)

);

/\*----------------------------------------------Usage\_History Table-------------------------------\*/

create table Usage\_History(

Usage\_Id int identity(1,1) primary key,

Customer\_Id int not null,

Bill\_Generated\_Date date not null,

Bill\_payment\_Date date not null,

Total\_Amount decimal(9,2),

Total\_Consumption decimal(9,2),

foreign key(Customer\_Id) references Customers(Customer\_Id)

);

/\*------------------------------------Insert values to Service\_Provider Table---------------------------\*/

use EMS;

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Reliant','8332359653','info@reliant.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Oncor','8263758293','info@oncor.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Compare Power','8462758364','info@comparepowwer.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Amigo Power','9162553475','info@Amigopower.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Ampra Energy','7253856495','info@amprapower.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Energy Texas','8371698365','info@energytexas.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Tara energy','7239476849','info@taraenergy.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Onver Power','5837469382','info@onverpower.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Tata Powers','8385945848','info@tatapower.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Cupa power','5837485737','info@cupapower.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('Zampa Distributions','4938572647','info@zampadistributions.com');

insert into service\_provider(Supplier\_Name,Supplier\_Contact,Supplier\_EMail)

values('spectrum powers','9472837485','info@tspectrumpowers.com');

/\*---------------------------------------Inserting values into Region table--------------------------------\*/

insert into Region values('Denton');

insert into Region values('Dallas');

insert into Region values('Houston');

insert into Region values('Austin');

insert into Region values('Irving');

insert into Region values('Plano');

insert into Region values('Frisco');

insert into Region values('Fort-worth');

insert into Region values('Lewisville');

insert into Region values('Phoenix');

/\*----------------------------------------Inserting values into Customers table---------------------------\*/

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Raghu','nandhan','4857362847','raghu@gmail.com','union circle, denton, tx',1,1,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('ram','charan','4837285674','ram@gmail.com','oaks of denton, denton, TX',3,1,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Email,Customer\_Contact,Customer\_Location,Supplier\_Id,Region\_Id)

values('Rishi','kumar','rishi@gmail.com','8573746385','valley ranch pkwy, Irving, TX',1,5);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('priya','chandran','9472847364','pricha@gmail.com','oaks at valley ranch, Irving, Tx',2,5,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Sruthi','Kiran','9263828462','kiran@gmail.com','Swadeshi plaza, Irving, TX',4,5,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Cristina','Rose','6284638472','cristina@gmail.com','Wilmington pkwy, Fort-worth, Tx',4,8,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Ronald','Wright','7483648264','ronald@gmail.com','Dallas pkwy, dallas, Tx',6,2,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Terissa','Tim','9273846284','teressa@gmail.com','Loscolinas pkwy, houston, Tx',5,3,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Mike','Tong','9374738472','Mike@gmail.com','Spring valley rd,plano, Tx',3,6,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Tom','Crissey','8284637463','Tom@gmail.com','Spring valley rd,plano, Tx',2,6,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Micheal','Tom','8264927482','Micheal@gmail.com','Montfort dr, frisco, Tx',1,7,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Pavan','Naini','9405959573','pavan@gmail.com','Ioof St, Houston, Tx',9,3,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Kalyan','Tinku','9890987890','tinku@gmail.com','Bernald St denton, Tx',1,1,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Kiran Kumar','Mintu','9309439890','kirankumar@gmail.com','lebin street, lw',7,9,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Adam','Smith','9453245679','adamsmith@gmail.com','Teasley Lane,Ph',7,10,12);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Anvar','mith','8293746738','anvar@gmail.com','Timber st, Lewisville, Tx',11,9,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('cane','walter','3947584839','cane@gmail.com','Twistley rd,Frisco,Tx',10,7,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Kane','county','5928385858','kane@gmail.com','winter ln, Plano, Tx',8,6,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Ruth','ranger','7838485868','ruth@gmail.com','Rinder pkwy, Dallas, Tx',6,2,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Birkey','wing','8273648573','birkey@gmail.com','oaks of denton, Denton, Tx',11,1,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('king','william','7394829485','kingwilliam@gmail.com','trails of austin, Austin,Tx',12,4,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('ram','singh','8483847275','ramsingh@gmail.com','lukes of fort-worth, Fort-worth, Tx',9,8,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Trint','bing','8938475837','trinth@gmail.com','churchy st, Phoenix, Tx',5,10,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('wright','wrong','8394388484','wright@gmail.com','trinty ln, Denton, Tx',8,1,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('miller','wright','8273645273','miller@gmail.com','west street ln, Irving, Tx',9,5,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Anil','kumbley','7483746574','kumbley@gmail.com','Timber west ln, Austin, Tx',2,4,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('virat','rona','7384958274','virat@gmail.com','valley pkwy E, Plano, Tx',4,6,1);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('vikranth','rana','8938485828','vikranth@gmail.com','ruster rd, Fort-worth, Tx',8,8,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('Ringer','rale','8394727475','ringer@gmail.com','winster pkwy, Austin, Tx',10,4,0);

insert into Customers(Customer\_FName,Customer\_LName,Customer\_Contact,Customer\_Email,Customer\_Location,Supplier\_Id,Region\_Id,Enrolledfor\_Autopayments)

values('stella','fring','9284837475','stella@gmail.com','oaks of Phoenix, Phoenix, Tx',11,10,0);

/\*--------------------------------Inserting values into Maintenance\_Requests Table-------------------\*/

insert into Maintenance\_Requests(Supplier\_Id,Maintenancerequest\_Date,[Description])

values(2,'2023-06-22','Maintenance is scheduled at the supplier end annually');

insert into Maintenance\_Requests

values(1,5,4,'2023-08-23','Request for Installation of meter');

insert into Maintenance\_Requests

values(1,1,1,'2023-06-12','Request for Installation of meter');

insert into Maintenance\_Requests(Supplier\_Id,Maintenancerequest\_Date,[Description])

values(5,'2023-08-11','Maintenance is scheduled at the supplier end annually');

insert into Maintenance\_Requests

values(4,8,6,'2023-04-23','Request for Installation of meter');

insert into Maintenance\_Requests

values(6,2,7,'2023-10-05','Request for Installation of meter');

insert into Maintenance\_Requests

values(5,3,8,'2023-09-02','Request for Installation of meter');

insert into Maintenance\_Requests

values(2,6,10,'2023-08-06','Request for Installation of meter');

insert into Maintenance\_Requests

values(1,7,11,'2023-10-01','Request for Installation of meter');

insert into Maintenance\_Requests

values(1,1,13,'2023-12-02','Request for Installation of meter');

insert into Maintenance\_Requests

values(7,10,15,'2023-11-04','Request for change of meter');

insert into Maintenance\_Requests

values(10,7,17,'2023-08-13','Request for Installation of meter');

insert into Maintenance\_Requests(Supplier\_Id,Maintenancerequest\_Date,[Description])

values(3,'2023-11-10','Maintenance is scheduled at the supplier end annually');

insert into Maintenance\_Requests

values(6,12,19,'2023-09-12','Request for change of meter');

insert into Maintenance\_Requests

values(11,1,20,'2023-08-13','Request for Installation of meter');

insert into Maintenance\_Requests

values(9,8,22,'2023-10-04','Request for change of meter');

insert into Maintenance\_Requests

values(5,10,23,'2023-11-13','Request for Installation of meter');

insert into Maintenance\_Requests

values(2,4,26,'2023-09-06','Request for change of meter');

insert into Maintenance\_Requests

values(8,8,28,'2023-08-03','Request for Installation of meter');

insert into Maintenance\_Requests

values(10,4,29,'2023-07-12','Request for Installation of meter');

/\*----------------------------------Inserting values into Meter\_Readings Table-------------------------\*/

insert into Meter\_Readings

values(1,23.56,'2023-11-20');

insert into Meter\_Readings

values(2,13.27,'2023-11-20');

insert into Meter\_Readings

values(3,20.46,'2023-11-20');

insert into Meter\_Readings

values(4,32.76,'2023-11-20');

insert into Meter\_Readings

values(5,53.2,'2023-11-20');

insert into Meter\_Readings

values(6,49.5,'2023-11-20');

insert into Meter\_Readings

values(7,42.3,'2023-11-20');

insert into Meter\_Readings

values(8,64.2,'2023-11-20');

insert into Meter\_Readings

values(9,40.3,'2023-11-20');

insert into Meter\_Readings

values(10,54.5,'2023-11-20');

insert into Meter\_Readings

values(11,46.7,'2023-11-20');

insert into Meter\_Readings

values(12,34.4,'2023-11-20');

insert into Meter\_Readings

values(13,10.1,'2023-11-20');

insert into Meter\_Readings

values(14,19.2,'2023-11-20');

insert into Meter\_Readings

values(15,43.2,'2023-11-20');

insert into Meter\_Readings

values(16,53.5,'2023-11-20');

insert into Meter\_Readings

values(17,23.27,'2023-11-20');

insert into Meter\_Readings

values(18,22.46,'2023-11-20');

insert into Meter\_Readings

values(19,22.76,'2023-11-20');

insert into Meter\_Readings

values(20,12.51,'2023-11-20');

insert into Meter\_Readings

values(21,33.56,'2023-11-20');

insert into Meter\_Readings

values(22,23.27,'2023-11-20');

insert into Meter\_Readings

values(23,25.46,'2023-11-20');

insert into Meter\_Readings

values(24,22.76,'2023-11-20');

insert into Meter\_Readings

values(25,16.51,'2023-11-20');

insert into Meter\_Readings

values(26,31.76,'2023-11-20');

insert into Meter\_Readings

values(27,19.27,'2023-11-20');

insert into Meter\_Readings

values(28,27.46,'2023-11-20');

insert into Meter\_Readings

values(29,37.76,'2023-11-20');

insert into Meter\_Readings

values(30,19.51,'2023-11-20');

/\*----------------------------------Inserting values into Auto\_Payments Table-------------------------\*/

insert into Auto\_Payments

values(1,'credit','4783927594716258','2024-02-01',346,'union circle');

insert into Auto\_Payments

values(4,'debit','8264857102859374','2025-12-01',153,'oaks at valley ranch');

insert into Auto\_Payments

values(6,'Debit','5263846273849270','2025-12-01',134,'Wilmington pkwy, Fort-worth, tx');

insert into Auto\_Payments

values(8,'debit','8263745261846370','2026-04-01',143,'Loscolinas pkwy, houston, tx');

insert into Auto\_Payments

values(11,'debit','8394728463746370','2024-03-01',256,'Montfort dr, frisco, tx');

insert into Auto\_Payments

values(12,'Credit','12345','2027-01-01',123,'Ioof st,tx');

insert into Auto\_Payments

values(14,'Debit','2341234564332','2030-11-01',533,'lebin street, lw');

insert into Auto\_Payments

values(15,'Credit','234567812234','2025-12-01',324,'bernard street,denton');

insert into Auto\_Payments

values(16,'credit','2150816150957480','2031-01-01',587,'Discovery park dr');

insert into Auto\_Payments

values(17,'credit','4838485969485860','2024-07-01',564,'Twistley rd,Frisco,Tx');

insert into Auto\_Payments

values(18,'debit','9483847562736450','2025-06-01',498,'winter ln, Plano, Tx');

insert into Auto\_Payments

values(23,'credit','2938485862949280','2024-08-01',734,'churchy st, Phoenix, Tx');

insert into Auto\_Payments

values(27,'debit','7374859384759280','2027-11-01',435,'valley pkwy E, Plano, Tx');

/\*----------------------------------Inserting values into Billing Table-------------------------\*/

insert into Billing

values(56.45,1,134.45,1,'2023-11-03');

insert into Billing

values(43.25,2,108.42,2,'2023-11-03');

insert into Billing

values(72.34,3,184.3,1,'2023-11-03');

insert into Billing

values(38.34,4,90.2,2,'2023-11-03');

insert into Billing

values(58.45,5,105.32,4,'2023-11-03');

insert into Billing

values(52.2,6,105.36,4,'2023-11-03');

insert into Billing

values(48.4,7,89.9,6,'2023-11-03');

insert into Billing

values(72.2,8,130.36,5,'2023-11-03');

insert into Billing

values(44.43,9,82.6,3,'2023-11-03');

insert into Billing

values(54.9,10,103.5,2,'2023-11-03');

insert into Billing

values(48.89,11,86.76,1,'2023-11-03');

insert into Billing

values(50.55,12,75,9,'2023-11-03');

insert into Billing

values(12,13,31,1,'2023-11-03');

insert into Billing

values(45,14,12,7,'2023-11-03');

insert into Billing

values(94,15,56,7,'2023-11-03');

insert into Billing

values(100,16,200,11,'2023-11-03');

insert into Billing

values(86.65,17,164.6,10,'2023-11-03');

insert into Billing

values(46.87,18,128.8,8,'2023-11-03');

insert into Billing

values(55.23,19,104.45,6,'2023-11-03');

insert into Billing

values(37.87,20,74.45,11,'2023-11-03');

insert into Billing

values(64.57,21,124.45,12,'2023-11-03');

insert into Billing

values(37.43,22,84.55,9,'2023-11-03');

insert into Billing

values(42.45,23,90.43,5,'2023-11-03');

insert into Billing

values(39.78,24,80.15,8,'2023-11-03');

insert into Billing

values(37.67,25,82.55,9,'2023-11-03');

insert into Billing

values(46.23,26,101.86,2,'2023-11-03');

insert into Billing

values(48.65,27,96.23,4,'2023-11-03');

insert into Billing

values(39.78,28,92.45,8,'2023-11-03');

insert into Billing

values(52.78,29,108.36,10,'2023-11-03');

insert into Billing

values(49.67,30,103.28,11,'2023-11-03');

/\*----------------------------------Inserting values into Reminders Table-------------------------\*/

insert into Reminders

VALUES('Your bill has paid usig autopayment method',1);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',2);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',3);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',4);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',5);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',6);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',7);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',8);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',9);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',10);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',11);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',12);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',13);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',14);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',15);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',16);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',17);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',18);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',19);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',20);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',21);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',22);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',23);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',24);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',25);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',26);

insert into Reminders

VALUES('Your bill has paid usig autopayment method',27);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',28);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',29);

insert into Reminders

VALUES('Your generated bill was due on 11/20/2023',30);

/\*----------------------------------Inserting values into Usage\_History Table-------------------------\*/

insert into Usage\_History

values(1,'2023-10-3','2023-10-18',56.78,126.54);

insert into Usage\_History

values(1,'2023-09-3','2023-09-15',53.18,106.54);

insert into Usage\_History

values(1,'11/3/2023','11/6/2023',46.77,92.5);

insert into Usage\_History

values(2,'2023-10-3','2023-10-18',46.78,88.54);

insert into Usage\_History

values(2,'11/3/2023','11/12/2023',36.77,72.5);

insert into Usage\_History

values(3,'11/3/2023','11/11/2023',33.7,66.9);

insert into Usage\_History

values(4,'11/3/2023','11/6/2023',20.3,40.6);

insert into Usage\_History

values(5,'2023-10-3','2023-10-18',36.3,70.1);

insert into Usage\_History

values(5,'11/3/2023','11/07/2023',36.5,70.2);

insert into Usage\_History

values(6,'11/3/2023','11/6/2023',33.6,67.3);

insert into Usage\_History

values(7,'11/3/2023','11/08/2023',29.6,60.4);

insert into Usage\_History

values(8,'11/3/2023','11/6/2023',39.5,80.4);

insert into Usage\_History

values(9,'11/3/2023','11/15/2023',60.3,130.4);

insert into Usage\_History

values(10,'11/3/2023','11/8/2023',53.5,104.3);

insert into Usage\_History

values(11,'2023-10-3','2023-06-18',44.7,95.6);

insert into Usage\_History

values(11,'11/3/2023','11/6/2023',50.4,100.8);

insert into Usage\_History

values(12,'2023-10-3','2023-06-18',39.5,80.9);

insert into Usage\_History

values(12,'2023-09-3','2023-06-15',37.8,75.8);

insert into Usage\_History

values(12,'11/3/2023','11/6/2023',41.7,90.2);

insert into Usage\_History

values(13,'11/3/2023','11/13/2023',33.33,66.66);

insert into Usage\_History

values(14,'11/3/2023','11/6/2023',42.27,82.7);

insert into Usage\_History

values(15,'2023-10-3','2023-06-18',52.5,102.54);

insert into Usage\_History

values(15,'11/3/2023','11/6/2023',37.77,82.54);

insert into Usage\_History

values(16,'11/3/2023','11/6/2023',66.77,122.5);

insert into Usage\_History

values(17,'2023-10-3','2023-06-18',37.5,60.5);

insert into Usage\_History

values(17,'2023-09-3','2023-06-15',35.9,69.9);

insert into Usage\_History

values(17,'11/3/2023','11/6/2023',35.7,69.8);

insert into Usage\_History

values(18,'11/3/2023','11/08/2023',37.9,77.8);

insert into Usage\_History

values(19,'2023-10-3','2023-07-18',41.5,92.4);

insert into Usage\_History

values(19,'11/3/2023','11/05/2023',34.9,74.3);

insert into Usage\_History

values(20,'11/3/2023','11/10/2023',44.4,88.8);

insert into Usage\_History

values(21,'11/3/2023','11/15/2023',28.6,52.4);

insert into Usage\_History

values(22,'11/3/2023','11/18/2023',34.8,74.6);

insert into Usage\_History

values(23,'2023-10-3','2023-06-18',39.6,80.3);

insert into Usage\_History

values(23,'2023-09-3','2023-06-15',36.7,71.4);

insert into Usage\_History

values(23,'11/3/2023','11/06/2023',38.9,72.4);

insert into Usage\_History

values(24,'2023-10-3','2023-12-18',47.4,90.3);

insert into Usage\_History

values(24,'11/3/2023','11/6/2023',55.9,112.5);

insert into Usage\_History

values(25,'2023-10-3','2023-10-18',44.8,90.5);

insert into Usage\_History

values(25,'2023-09-3','2023-09-15',51.7,103.2);

insert into Usage\_History

values(25,'11/3/2023','11/12/2023',41.7,83.4);

insert into Usage\_History

values(26,'2023-10-3','2023-11-18',37.6,72.4);

insert into Usage\_History

values(26,'11/3/2023','11/13/2023',33.6,66.7);

insert into Usage\_History

values(27,'2023-10-3','2023-06-18',61.2,120.5);

insert into Usage\_History

values(27,'2023-09-3','2023-06-15',41.4,83.2);

insert into Usage\_History

values(27,'11/3/2023','11/06/2023',39.5,76.8);

insert into Usage\_History

values(28,'2023-10-3','2023-08-18',53.4,101.2);

insert into Usage\_History

values(28,'11/3/2023','11/12/2023',28.9,60.2);

insert into Usage\_History

values(29,'2023-10-3','2023-09-18',36.7,72.2);

insert into Usage\_History

values(29,'11/3/2023','11/15/2023',54.6,102.2);

insert into Usage\_History

values(30,'2023-10-3','2023-05-18',45.38,90.5);

insert into Usage\_History

values(30,'11/3/2023','11/06/2023',43.34,86.9);