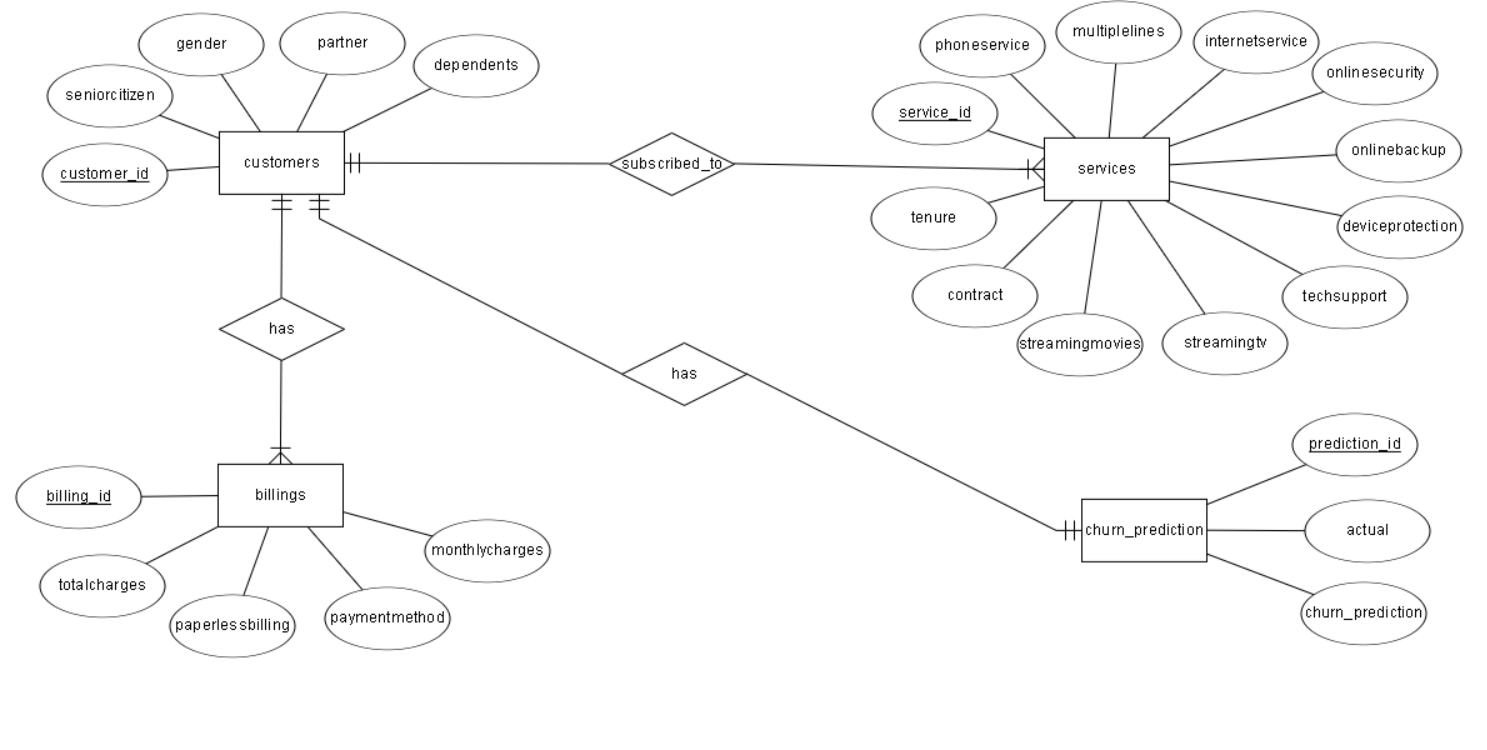
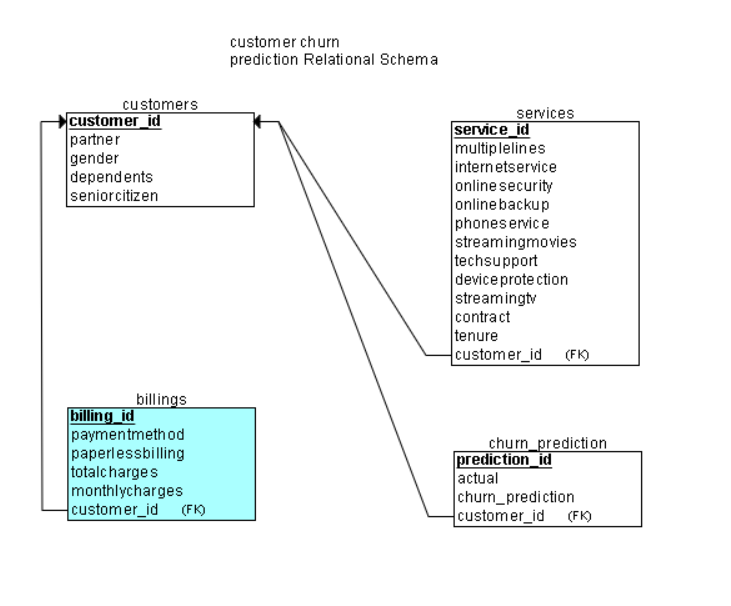
**Module 3: SQL REPORT**

**Database Schema Design:**

**ERD Diagram for Customer Churn Prediction:**

****

**Relational Schema:**

****

**Creation of database and tables through sql queries:**

-- creating the database for customer churn prediction

create database customer\_churn;

-- using that database

use customer\_churn;

CREATE TABLE customers (

customer\_id VARCHAR(50) NOT NULL,

partner ENUM('No', 'Yes') NOT NULL,

gender ENUM('Male', 'Female') NOT NULL,

dependents ENUM('No', 'Yes') NOT NULL,

seniorcitizen ENUM('No', 'Yes') NOT NULL,

PRIMARY KEY (customer\_id)

);

CREATE TABLE services (

service\_id INT NOT NULL AUTO\_INCREMENT,

multiplelines ENUM('No', 'Yes', 'No phone service') NOT NULL,

internetservice ENUM('Fiber optic', 'DSL', 'No') NOT NULL,

onlinesecurity ENUM('No', 'Yes', 'No internet service') NOT NULL,

onlinebackup ENUM('No', 'Yes', 'No internet service') NOT NULL,

phoneservice ENUM('No', 'Yes') NOT NULL,

streamingmovies ENUM('No', 'Yes', 'No internet service') NOT NULL,

techsupport ENUM('No', 'Yes', 'No internet service') NOT NULL,

deviceprotection ENUM('No', 'Yes', 'No internet service') NOT NULL,

streamingtv ENUM('No', 'Yes', 'No internet service') NOT NULL,

contract ENUM('Monthly', 'Yearly') NOT NULL,

tenure INT NOT NULL,

customer\_id VARCHAR(50) NOT NULL,

PRIMARY KEY (service\_id),

CONSTRAINT fk\_services\_customer\_id FOREIGN KEY (customer\_id) REFERENCES customers(customer\_id)

);

CREATE TABLE billings (

billing\_id INT NOT NULL AUTO\_INCREMENT,

paymentmethod ENUM('Electronic check', 'Mailed check', 'Bank transfer (automatic)', 'Credit card (automatic)') NOT NULL,

paperlessbilling ENUM('No', 'Yes') NOT NULL,

totalcharges FLOAT NOT NULL,

monthlycharges FLOAT NOT NULL,

customer\_id VARCHAR(50) NOT NULL,

PRIMARY KEY (billing\_id),

CONSTRAINT fk\_billings\_customer\_id FOREIGN KEY (customer\_id) REFERENCES customers(customer\_id)

);

CREATE TABLE churn\_prediction (

prediction\_id INT NOT NULL AUTO\_INCREMENT,

actual BIT NOT NULL,

churn\_prediction BIT NOT NULL,

customer\_id VARCHAR(50) NOT NULL,

PRIMARY KEY (prediction\_id),

CONSTRAINT fk\_churn\_prediction\_customer\_id FOREIGN KEY (customer\_id) REFERENCES customers(customer\_id)

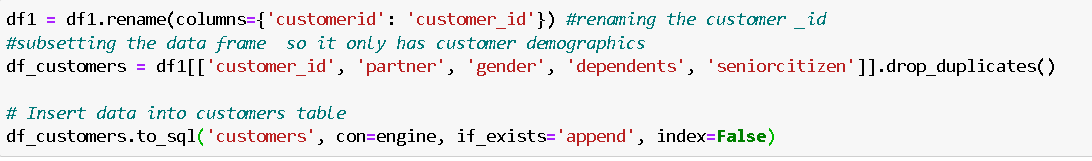
);

**Data Ingestion:**

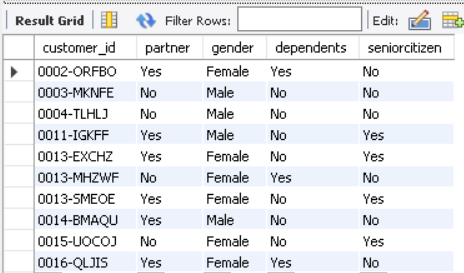
**Creating a Connection:**



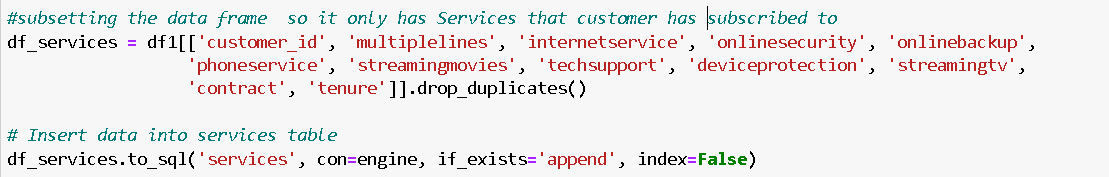
**Inserting data in to Customers table:**



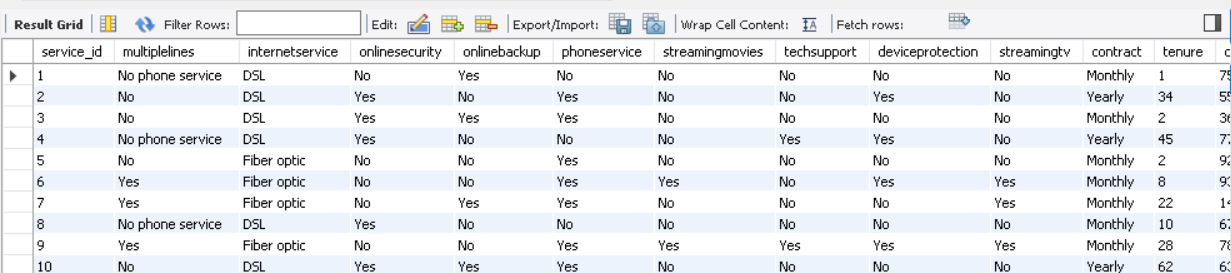
**Verifying through sql:**

****

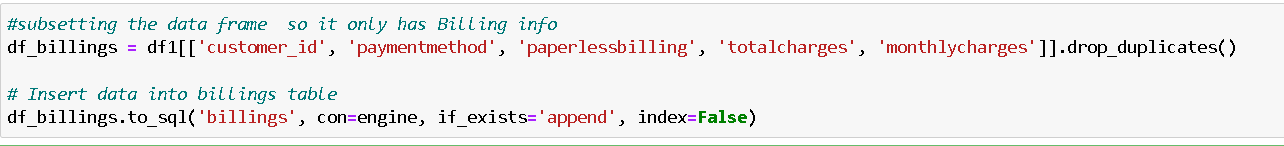
**Inserting data in to Services table:**

****

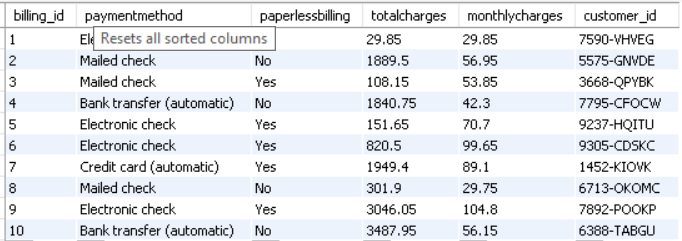
**Verifying through sql:**

****

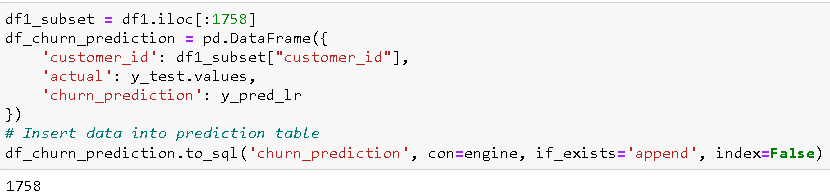
**Inserting data in to billings table:**

****

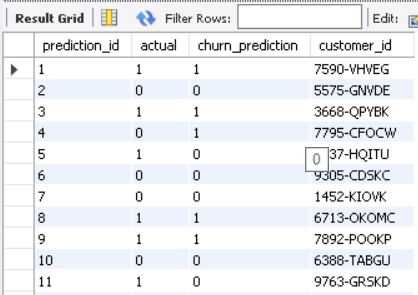
**Verifying through sql:**

****

**Inserting data in to prediction table:**

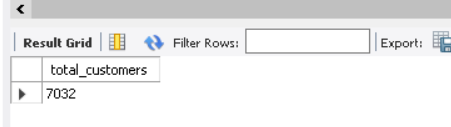
****

**Verifying through sql:**

****

**Querying through Data :**

**-- finding out the total number of customers**

select count(\*) as total\_customers from customers;

**-- finding out the average tenure**

select avg(tenure) as avg\_tenure from services;

**-- finding out the total revenue**

select sum(totalcharges) as total\_revenue from billings;



**-- displaying customers that have a yearly contract**

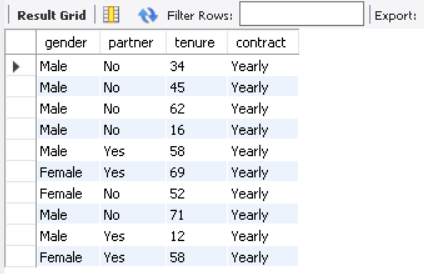
select gender,partner ,tenure,contract

from customers c

inner join services s

on c.customer\_id=s.customer\_id

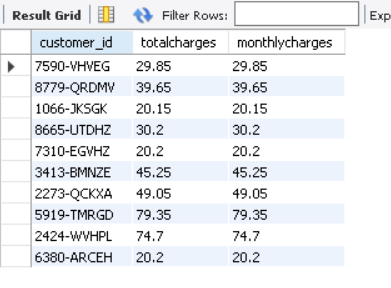
where contract="Yearly";



**-- Displaying customers with the equal total and monthly charges**

select customer\_id,totalcharges,monthlycharges from

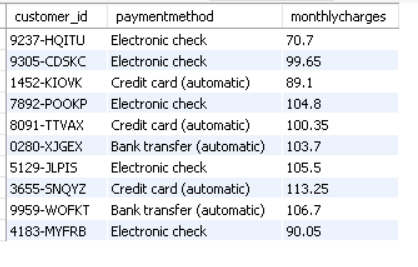
billings where monthlycharges = totalcharges;



**-- Displaying customers with the monthly charges greater than $70**

select customer\_id,paymentmethod ,monthlycharges from

billings where monthlycharges >70.0;



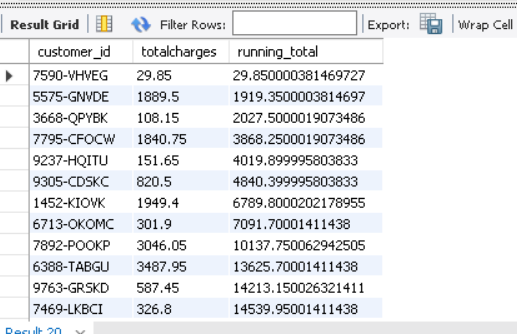
**Advanced SQL Analysis:**

**-- Calulcating the cummulative sum of total charges**

select customer\_id, totalcharges,

sum(totalcharges) over (order by billing\_id) as running\_total

from billings;



**-- Displaying average of monthly charges over last 5 months**

select customer\_id, monthlycharges,

avg(monthlycharges) over (order by billing\_id rows between 5 preceding and current row) as moving\_average

from billings;

