E-Invest Inc.

By:

Raymond Delgado - Senior Oracle Database Administrator, Project Leader

Savita Yadav - Database Developer

Davide Tita - Database Analyst

Ninelia Talverdi - Database QA Officer

CIS 5430 - Databases and Data Warehousing

Dr. Ming Wang

College of Business and Economics

California State University, Los Angeles

May 2021

Contents

Introduction

E-Invest is an investment company that has a new stock-mutual fund trading platform which keeps track of investments of clients who invest money into stocks and mutual funds. The unique characteristic of the E-Invest platform is that it offers its services to both individuals and investing firms.

This documentation shows how the company keeps track of the clients' investments through the development and implementation of an Oracle database developed on Oracle environment. Each client has a unique taxpayer identification number. A client can be an individual or a business that may invest in stocks or mutual funds.

The E-Invest database system has various functionalities and it includes 8 tables in a relational schema. The stocks and mutual funds are identified by unique tickers and the tables include current prices, annual highest/lowest prices and yearly returns. Moreover, ratings are assigned to assess the investment (i.e. A, B, C).

The database technician can use data manipulation language (DML) to insert, delete, and update records and also use various features developed by PL/SQL to automate price increases to stock or mutual funds. The database also performs advanced input validation. For instance, it will not accept zero as a valid current price for mutual funds.

Part 1 – Database Design - Questions 1 to 7

1. Extract Entity Types and Identify Relationship Matrix

Based on the given Investment Portfolio Scenario in Section 3.1, four external entity types have been extracted. They are identified as: Client, Stock, Mutual Fund and Family Fund

Among the four entity types, three pairs of relationships have been identified in the following relationship matrix.

	Client	**Stock**	**Mutual Fund**	**Family Fund**
Client	--	invests in	invests into	--
Stock	bought by	--		
Mutual Fund	bought by	--	--	belongs to
Family Fund	--	--	have	--

Table 1 - Relationship Matrix of the E- Invest Inc.

2. Generated Business Rules

The three pairs of relationships are further defined as the database business rules as follows:

One client may invest in one or more than one stock.
A stock may be bought by one or more than one client.

One client may invest in one or more than one mutual fund.
A mutual fund may be bought by one or more than one client.

One fund family must have one or more than one mutual fund. One fund must belong to only one fund family.

3. Entity-Relationship Diagram (ERD)

Based on the above identified entities and relationships, the E-Invest Inc. entity-relationship diagram is developed using the Crow's Foot Database Notation.

1) entity types, 2) relationship types, 3) keys, 4) cardinality and 5) PKs and FKs

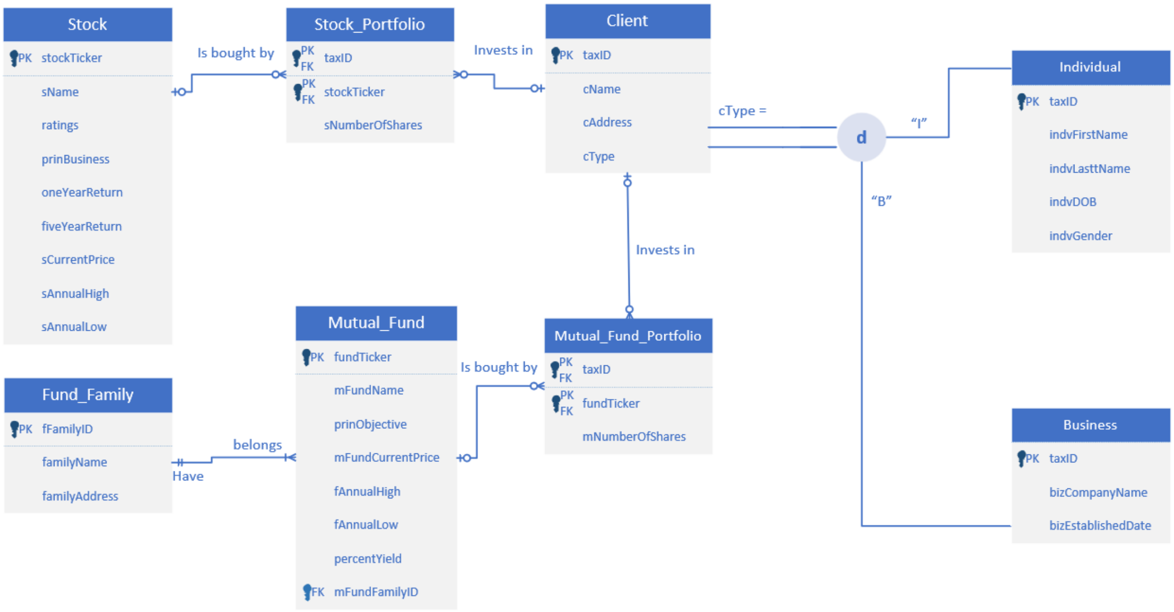


Figure 1 - Entity Relationship Diagram (ERD) of the E- Invest Inc.

4. Database Logical Design

Eight relational tables are mapped from the above ER diagram. Each table has a table name, a primary key (PK), a foreign key(FK), and its corresponding additional fields.

Table Name: Client

taxID (PK)	cName	cAddress

*cAddress(streetName, cityName, zipCode)

Table Name: Stock

stockTicker (PK)	sName	ratings	prinBusiness	sCurrentPrice
sAnnualHigh	sAnnualLow	oneYearReturn	fiveYearReturn	

Table Name: Mutual_Fund

fundTicker (PK)	mFundName	prinObjective	mFundCurrentPrice	mFundAnnualHigh
mFundAnnualLow	percentYield	mFundFamilyID (FK)		

Table Name: Fund_Family

fFamilyID (PK)	familyName	familyAddress

*familyAddress(streetName, cityName, zipCode)

Table Name: Stock_Portfolio

taxID (PK) (FK1)	stockTicker (PK) (FK2)	sNumberOfShares

Table Name: Mutual_Fund_Portfolio

taxID (PK) (FK1)	fundTicker(PK) (FK2)	mFundNumberOfShares

Table Name: Individual

taxID (PK) (FK1)	indvFirstName	indvLastName	indvDOB	indvGender

*indvAddress(streetName, cityName, zipCode)

Table Name: Business

taxID (PK) (FK1)	bizCompanyName	bizEstablishedDate

*bizAddress(streetName, cityName, zipCode)

5. Establish Referential integrity

The following referential integrities are established in order to relate the tables to each other in the database. Each of the referential integrities shows that a foreign key refers to the corresponding primary key. F.K.-> P.K.

- Stock_Portfolio.taxId (FK) -> Client.taxId (PK)
- Stock_Portfolio.stockTicker (FK) -> Stock.stockTicker (PK)
- Mutual_Fund_Portfolio.taxID (FK) -> Client.taxID (PK)
- Mutual_Fund_Portfolio.fundTicker (FK) -> Mutual_Fund.fundTicker (PK)
- Mutual_Fund.mFundFamilyId (FK) -> Fund_Family.fFamilyId (PK)
- Individual.taxID (FK) -> Client.taxID (PK)
- Business.taxID (FK) -> Client.taxID (PK)

6. Function analysis for each of the tables

Attribute A -> Attribute B (Determinant attribute(s) Determines Dependent Attribute(s))

Table: Client
TaxID → cName, cAddress (Full)

Table: Stock
StockTicker → sName, Ratings, prinBusiness, sCurrentPrice, sAnnualHigh, sAnnualLow, oneYearReturn, fiveYearReturn (Full)

Table: Mutual_Fund
FundTicker → mFundName, prinObjective, mFundCurrentPrice, mFundAnnualHigh, mFundAnnualLow, percentYield (Full)

Table: Fund_Family
fFamilyID → familyName, familyAddress(Full)

Table: Stock_Portfolio
TaxID, StockTicker → sNumberOfShares (Full)

Mutual_Fund_Portfolio
TaxID, FundTicker → mFundNumberOfShares (Full)

Table: Individual
TaxID → indvFirstName,indvLastname,indvDOB,indvGender (Full)

Table: Business
TaxID → bizCompanyName, bizEstablishedDate (Full)

7. Show all the normalized tables and indicate the normalization form for each of the tables

All tables are normalized to the third normal form. Neither partial dependencies nor transitive dependencies are presented in any of the six tables.

Table Name	1NF	2NF	3NF
Client	X	X	X
Stock	X	X	X
Mutual_Fund	X	X	X
Fund_Family	X	X	X
Stock_Portfolio	X	X	X
Mutual_Fund_Portfolio	X	X	X
Individual	X	X	X
Business	X	X	X

Table 2 - Normalization of the E- Invest Inc.

Part 2 - Database Development with SQL in Oracle DBMS - Questions 4 to 9

Database creation with constraints and their output

4. Create the relational database with five constraints in appropriate tables and load data into the database via ORACLE SQL*PLUS.

5. Printout of SQL DDL script (Database creation script), database structure/relational schema (DESC TableName) and database instance (SELECT * FROM TableName).

Object creation script

```
--Create Object Address
CREATE OR REPLACE TYPE address_ty AS OBJECT [Raymond - Object]
(streetName VARCHAR2(50),
 cityName VARCHAR2(20),
 zipCode VARCHAR2(5));
```

1. Client Table

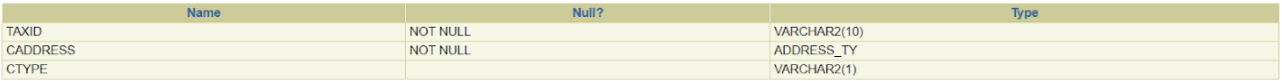
Database creation script

```
--Create Table Client
CREATE TABLE Client
(TaxID VARCHAR2(10) NOT NULL,
cAddress address_ty NOT NULL,
cType VARCHAR2(1) CHECK (cType IN ('I', 'B')),
CONSTRAINT client_taxID_pk PRIMARY KEY (taxID));
```

Database structure/relational schema

```
--Describe Table Client
DESC Client;
```

Name	Null?	Type
TAXID	NOT NULL	VARCHAR2(10)
CADDRESS	NOT NULL	ADDRESS_TY
CTYPE		VARCHAR2(1)



Name	Null?	Type
TAXID	NOT NULL	VARCHAR2(10)
CADDRESS	NOT NULL	ADDRESS_TY
CTYPE		VARCHAR2(1)

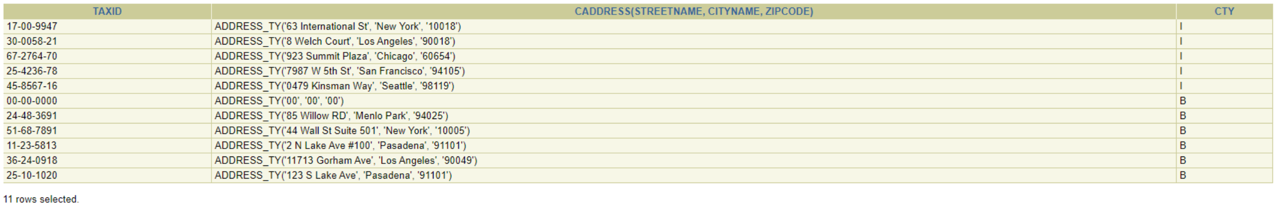
Database insert statement

```
--Insert into Client Table(I)
INSERT INTO Client values
('17-00-9947',address_ty('63 International St','New York','10018'), 'I');
INSERT INTO Client values
('30-0058-21',address_ty('8 Welch Court','Los Angeles','90018'), 'I');
INSERT INTO Client values
('67-2764-70',address_ty('923 Summit Plaza','Chicago','60654'), 'I');
INSERT INTO Client values
('25-4236-78',address_ty('7987 W 5th St','San Francisco','94105'), 'I');
INSERT INTO Client values
('45-8567-16',address_ty('0479 Kinsman Way','Seattle','98119'), 'I');

--Insert into Client(B)
INSERT INTO Client values
('00-00-0000', address_ty('00', '00', '00'),'B');
INSERT INTO Client values
('24-48-3691', address_ty('85 Willow RD', 'Menlo Park', '94025'), 'B');
INSERT INTO Client values
('51-68-7891', address_ty('44 Wall St Suite 501', 'New York', '10005'), 'B');
INSERT INTO Client values
('11-23-5813', address_ty('2 N Lake Ave #100', 'Pasadena', '91101'), 'B');
INSERT INTO Client values
('36-24-0918', address_ty('11713 Gorham Ave', 'Los Angeles', '90049'), 'B');
INSERT INTO Client values
('25-10-1020', address_ty('123 S Lake Ave', 'Pasadena', '91101'), 'B');
```

Database instance

```
--Select * from Table Client
SELECT *
FROM Client;
```

TAXID	CADDRESS(STREETNAME, CITYNAME, ZIPCODE)	CTY
17-00-9947	ADDRESS_TY('63 International St', 'New York', '10018')	I
30-0058-21	ADDRESS_TY('8 Welch Court', 'Los Angeles', '90018')	I
67-2764-70	ADDRESS_TY('923 Summit Plaza', 'Chicago', '60654')	I
25-4236-78	ADDRESS_TY('7987 W 5th St', 'San Francisco', '94105')	I
45-8567-16	ADDRESS_TY('0479 Kinsman Way', 'Seattle', '98119')	I
00-00-0000	ADDRESS_TY('00', '00', '00')	B
24-48-3691	ADDRESS_TY('85 Willow RD', 'Menlo Park', '94025')	B
51-68-7891	ADDRESS_TY('44 Wall St Suite 501', 'New York', '10005')	B
11-23-5813	ADDRESS_TY('2 N Lake Ave #100', 'Pasadena', '91101')	B
36-24-0918	ADDRESS_TY('11713 Gorham Ave', 'Los Angeles', '90049')	B
25-10-1020	ADDRESS_TY('123 S Lake Ave', 'Pasadena', '91101')	B



TAXID	CADDRESS(STREETNAME, CITYNAME, ZIPCODE)	CTY
17-00-9947	ADDRESS_TY('63 International St', 'New York', '10018')	I
30-0058-21	ADDRESS_TY('8 Welch Court', 'Los Angeles', '90018')	I
67-2764-70	ADDRESS_TY('923 Summit Plaza', 'Chicago', '60654')	I
25-4236-78	ADDRESS_TY('7987 W 5th St', 'San Francisco', '94105')	I
45-8567-16	ADDRESS_TY('0479 Kinsman Way', 'Seattle', '98119')	I
00-00-0000	ADDRESS_TY('00', '00', '00')	B
24-48-3691	ADDRESS_TY('85 Willow RD', 'Menlo Park', '94025')	B
51-68-7891	ADDRESS_TY('44 Wall St Suite 501', 'New York', '10005')	B
11-23-5813	ADDRESS_TY('2 N Lake Ave #100', 'Pasadena', '91101')	B
36-24-0918	ADDRESS_TY('11713 Gorham Ave', 'Los Angeles', '90049')	B
25-10-1020	ADDRESS_TY('123 S Lake Ave', 'Pasadena', '91101')	B

11 rows selected.

Object creation script

```
--Create Object Name in Individual Clients [Ninelia - Object]
CREATE OR REPLACE TYPE name_ty AS OBJECT
(indvFirstName VARCHAR2(20),
 indvLastName VARCHAR2(20));

--Create Object Date Of Birth and Gender in Individual Clients [Savita - Object]
CREATE OR REPLACE TYPE personal_info_ty as OBJECT
(indvDOB DATE,
 indvGender VARCHAR2(1));
```

2. Individual Table

Database creation script

```
--Create Table Individual
CREATE TABLE Individual
(taxID VARCHAR2 (10) NOT NULL,
indvName name_ty NOT NULL,
indvInformation personal_info_ty CHECK (indvInformation.indvGender IN ('M', 'F', 'O')),
CONSTRAINT individual_taxID_pk PRIMARY KEY (taxID));
```

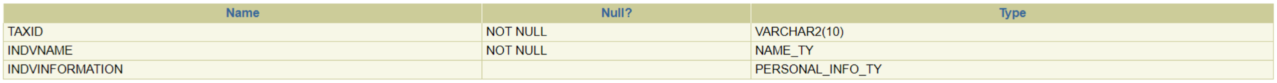
Add foreign keys

```
--Add Foreign Keys to Individual
ALTER TABLE Individual
ADD FOREIGN KEY (taxID) REFERENCES Client(taxID);
```

Database structure/relational schema

```
--Describe Table Individual
DESC Individual;
```

Name	Null?	Type
TAXID	NOT NULL	VARCHAR2(10)
INDVNAME	NOT NULL	NAME_TY
INDVINFORMATION		PERSONAL_INFO_TY



Name	Null?	Type
TAXID	NOT NULL	VARCHAR2(10)
INDVNAME	NOT NULL	NAME_TY
INDVINFORMATION		PERSONAL_INFO_TY

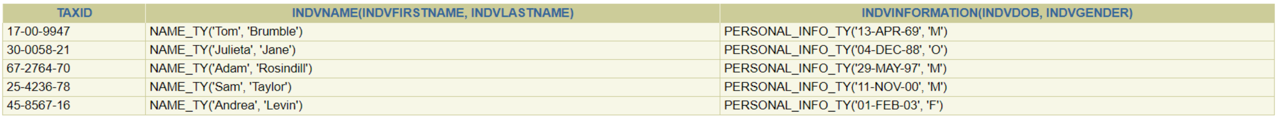
Database insert statement

```
--Insert into Individual Table
INSERT INTO Individual values
('17-00-9947', name_ty('Tom','Brumble'), personal_info_ty('13-APR-1969', 'M'));
INSERT INTO Individual values
('30-0058-21',name_ty('Julieta' ,'Jane'),personal_info_ty('04-DEC-1988','O'));
INSERT INTO Individual values
('67-2764-70',name_ty('Adam', 'Rosindill'), personal_info_ty('29-MAY-1997','M'));
INSERT INTO Individual values
('25-4236-78',name_ty('Sam', 'Taylor'), personal_info_ty('11-NOV-2000','M'));
INSERT INTO Individual values
('45-8567-16',name_ty('Andrea', 'Levin'), personal_info_ty('01-FEB-2003', 'F'));
```

Database instance

```
--Select * from Table Individual
SELECT *
FROM Individual;
```

TAXID	INDVNAME(INDVFIRSTNAME, INDVLASTNAME)	INDVINFORMATION(INDVDOB, INDVGENDER)
17-00-9947	NAME_TY('Tom', 'Brumble')	PERSONAL_INFO_TY('13-APR-69', 'M')
30-0058-21	NAME_TY('Julieta', 'Jane')	PERSONAL_INFO_TY('04-DEC-88', 'O')
67-2764-70	NAME_TY('Adam', 'Rosindill')	PERSONAL_INFO_TY('29-MAY-97', 'M')
25-4236-78	NAME_TY('Sam', 'Taylor')	PERSONAL_INFO_TY('11-NOV-00', 'M')
45-8567-16	NAME_TY('Andrea', 'Levin')	PERSONAL_INFO_TY('01-FEB-03', 'F')



TAXID	INDVNAME(INDVFIRSTNAME, INDVLASTNAME)	INDVINFORMATION(INDVDOB, INDVGENDER)
17-00-9947	NAME_TY('Tom', 'Brumble')	PERSONAL_INFO_TY('13-APR-69', 'M')
30-0058-21	NAME_TY('Julieta', 'Jane')	PERSONAL_INFO_TY('04-DEC-88', 'O')
67-2764-70	NAME_TY('Adam', 'Rosindill')	PERSONAL_INFO_TY('29-MAY-97', 'M')
25-4236-78	NAME_TY('Sam', 'Taylor')	PERSONAL_INFO_TY('11-NOV-00', 'M')
45-8567-16	NAME_TY('Andrea', 'Levin')	PERSONAL_INFO_TY('01-FEB-03', 'F')

3. Business Table

Database creation script

```
--Create Table Business
CREATE TABLE Business
(taxID VARCHAR2 (10) NOT NULL,
 bizCompanyName VARCHAR2(40) NOT NULL,
bizEstablishedDate Date,
CONSTRAINT business_taxID_pk PRIMARY KEY (taxID));
```

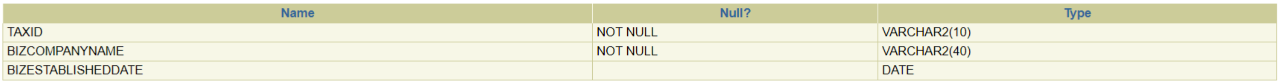
Add foreign keys

```
--Add Foreign Keys to Business
ALTER TABLE Business
ADD FOREIGN KEY (taxID) REFERENCES Client(taxID);
```

Database structure/relational schema

```
--Describe Table Business
DESC Business;
```

Name	Null?	Type
TAXID	NOT NULL	VARCHAR2(10)
BIZCOMPANYNAME	NOT NULL	VARCHAR2(40)
BIZESTABLISHEDDATE		DATE



Name	Null?	Type
TAXID	NOT NULL	VARCHAR2(10)
BIZCOMPANYNAME	NOT NULL	VARCHAR2(40)
BIZESTABLISHEDDATE		DATE

Database insert statement

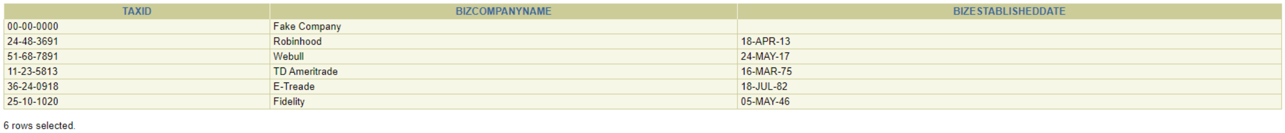
```
--Insert into Business Table
INSERT INTO Business values
('00-00-0000', 'Fake Company',NULL);
INSERT INTO Business values
('24-48-3691', 'Robinhood', '18-APR-2013');
INSERT INTO Business values
('51-68-7891', 'Webull', '24-MAY-2017');
INSERT INTO Business values
('11-23-5813', 'TD Ameritrade', '16-MAR-1975');
INSERT INTO Business values
```

```
('36-24-0918', 'E-Treade', '18-JUL-1982');
INSERT INTO Business values
('25-10-1020', 'Fidelity', '05-MAY-1946');
```

Database instance

```
--Select * from Table Business
SELECT *
FROM Business;
```

TAXID	BIZCOMPANYNAME	BIZESTABLISHEDDATE
24-48-3691	Robinhood	18-APR-13
51-68-7891	Webull	24-MAY-17
11-23-5813	TD Ameritrade	16-MAR-75
36-24-0918	E-Treade	18-JUL-82
25-10-1020	Fidelity	05-MAY-46



TAXID	BIZCOMPANYNAME	BIZESTABLISHEDDATE
00-00-0000	Fake Company	
24-48-3691	Robinhood	18-APR-13
51-68-7891	Webull	24-MAY-17
11-23-5813	TD Ameritrade	16-MAR-75
36-24-0918	E-Treade	18-JUL-82
25-10-1020	Fidelity	05-MAY-46

6 rows selected.

4. Stock Table

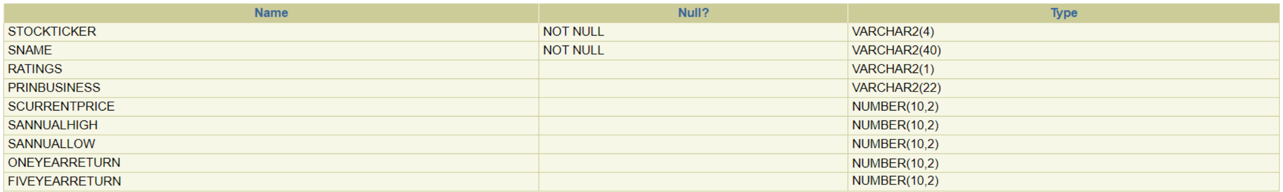
Database creation script

```
--Create Table Stock
CREATE TABLE Stock
(stockTicker VARCHAR2(4) NOT NULL,
sName VARCHAR2(40) NOT NULL UNIQUE,
ratings VARCHAR2(1) CHECK (ratings IN ('A', 'B', 'C', 'D', 'F')),
prinBusiness VARCHAR2(22),
sCurrentPrice NUMBER(10,2),
sAnnualHigh NUMBER(10,2),
sAnnualLow NUMBER(10,2),
oneYearReturn NUMBER (10,2),
fiveYearReturn NUMBER(10,2),
CONSTRAINT stock_stockTicker_pk PRIMARY KEY (stockTicker));
```

Database structure/relational schema

```
--Describe Table Stock
DESC Stock;
```

Name	Null?	Type
STOCKTICKER	NOT NULL	VARCHAR2(4)
SNAME	NOT NULL	VARCHAR2(40)
RATINGS		VARCHAR2(1)
PRINBUSINESS		VARCHAR2(22)
SCURRENTPRICE		NUMBER(10,2)
SANNUALHIGH		NUMBER(10,2)
SANNUALLOW		NUMBER(10,2)
ONEYEARRETURN		NUMBER(10,2)
FIVEYEARRETURN		NUMBER(10,2)



Name	Null?	Type
STOCKTICKER	NOT NULL	VARCHAR2(4)
SNAME	NOT NULL	VARCHAR2(40)
RATINGS		VARCHAR2(1)
PRINBUSINESS		VARCHAR2(22)
SCURRENTPRICE		NUMBER(10,2)
SANNUALHIGH		NUMBER(10,2)
SANNUALLOW		NUMBER(10,2)
ONEYEARRETURN		NUMBER(10,2)
FIVEYEARRETURN		NUMBER(10,2)

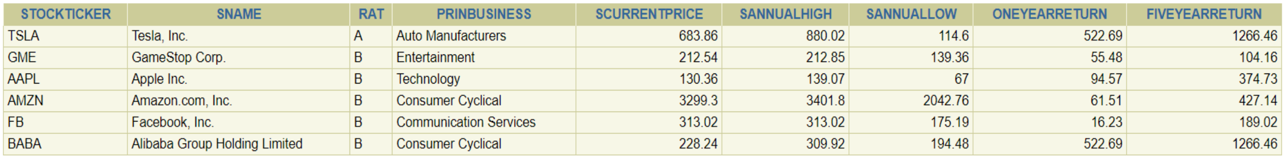
Database insert statement

```
--Insert into Table Stock
INSERT INTO Stock values
('TSLA', 'Tesla, Inc.', 'A', 'Auto Manufacturers', 683.86, 880.02, 114.60, 522.69, 1266.46);
INSERT INTO Stock values
('GME', 'GameStop Corp.', 'B', Entertainment, 212.54, 212.85, 139.36, 55.48, 104.16);
INSERT INTO Stock values
('AAPL', 'Apple Inc.', 'B', 'Technology', 130.36, 139.07, 67, 94.57, 374.73);
INSERT INTO Stock values
('AMZN', 'Amazon.com, Inc.', 'B', 'Consumer Cyclical', 3299.30, 3401.80, 2042.76, 61.51, 427.14);
INSERT INTO Stock values
('FB', 'Facebook, Inc.', 'B', 'Communication Services', 313.02, 313.02, 175.19, 16.23, 189.02);
INSERT INTO Stock values
('BABA', 'Alibaba Group Holding Limited', 'B', 'Consumer Cyclical', 228.24, 309.92, 194.48, 522.69,
1266.46);
```

Database instance

```
--Select * from Table Stock
SELECT *
FROM Stock;
```

STOCKTICKER	SNAME	RAT	PRINBUSINESS	SCURRENTPRICE	SANNUALHIGH	SANNUALLOW	ONEYEARRETURN	FIVEYEARRETURN
TSLA	Tesla, Inc.	A	Auto Manufacturers	683.86	880.02	114.6	522.69	1266.46
GME	GameStop Corp.	B	Technology	212.54	212.85	139.36	55.48	104.16
AAPL	Apple Inc.	B	Technology	130.36	139.07	67	94.57	374.73
AMZN	Amazon.com, Inc.	B	Consumer Cyclical	3299.3	3401.8	2042.76	61.51	427.14
FB	Facebook, Inc.	B	Communication Services	313.02	313.02	175.19	16.23	189.02
BABA	Alibaba Group Holding Limited	B	Consumer Cyclical	228.24	309.92	194.48	522.69	1266.46



STOCKTICKER	SNAME	RAT	PRINBUSINESS	SCURRENTPRICE	SANNUALHIGH	SANNUALLOW	ONEYEARRETURN	FIVEYEARRETURN
TSLA	Tesla, Inc.	A	Auto Manufacturers	683.86	880.02	114.6	522.69	1266.46
GME	GameStop Corp.	B	Entertainment	212.54	212.85	139.36	55.48	104.16
AAPL	Apple Inc.	B	Technology	130.36	139.07	67	94.57	374.73
AMZN	Amazon.com, Inc.	B	Consumer Cyclical	3299.3	3401.8	2042.76	61.51	427.14
FB	Facebook, Inc.	B	Communication Services	313.02	313.02	175.19	16.23	189.02
BABA	Alibaba Group Holding Limited	B	Consumer Cyclical	228.24	309.92	194.48	522.69	1266.46

5. Fund_Family Table

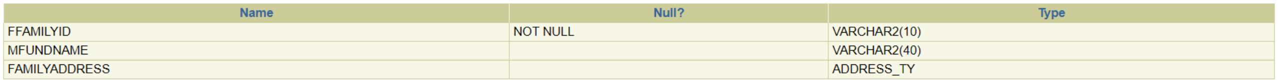
Database creation script

```
--Create Table Fund_Family  [Davide - Object]
CREATE TABLE Fund_Family
(fFamilyID VARCHAR2(10) NOT NULL,
 mFundName VARCHAR2(40),
familyAddress address_ty);


--Add Primary Key to Fund_Family Table
ALTER TABLE Fund_Family
ADD CONSTRAINT fund_family_pk PRIMARY KEY(fFamilyID);
```

Database structure/relational schema

```
--Describe Table Fund_Family
DESC Fund_Family;
```

Name	Null?	Type
FFAMILYID	NOT NULL	VARCHAR2(10)
MFUNDNAME		VARCHAR2(40)
FAMILYADDRESS		ADDRESS_TY



Name	Null?	Type
FFAMILYID	NOT NULL	VARCHAR2(10)
MFUNDNAME		VARCHAR2(40)
FAMILYADDRESS		ADDRESS_TY

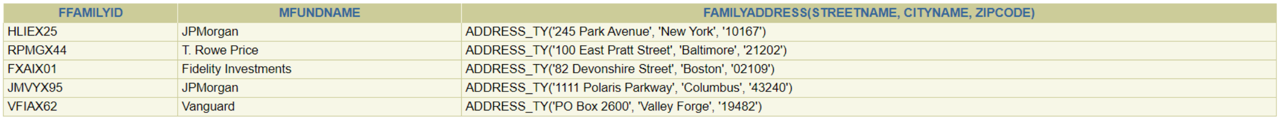
Database insert statement

```
--Insert into Table Fund_Family
INSERT INTO Fund_Family values
('HLIEX25','JPMorgan', address_ty('245 Park Avenue', 'New York', '10167'));
INSERT INTO Fund_Family values
('RPMGX44','T. Rowe Price', address_ty('100 East Pratt Street', 'Baltimore', '21202'));
INSERT INTO Fund_Family values
('FXAIX01','Fidelity Investments', address_ty('82 Devonshire Street', 'Boston', '02109'));
INSERT INTO Fund_Family values
('JMVYX95','JPMorgan', address_ty('1111 Polaris Parkway', 'Columbus', '43240'));
INSERT INTO Fund_Family values
('VFIAX62','Vanguard', address_ty('PO Box 2600', 'Valley Forge', '19482'));
```

Database instance

```
--Select * from Table Fund_Family
SELECT *
FROM Fund_Family;
```

FFAMILYID	MFUNDNAME	FAMILYADDRESS(STREETNAME, CITYNAME, ZIPCODE)
HLIEX25	JPMorgan	ADDRESS_TY('245 Park Avenue', 'New York', '10167')
RPMGX44	T. Rowe Price	ADDRESS_TY('100 East Pratt Street', 'Baltimore', '21202')
FXAIX01	Fidelity Investments	ADDRESS_TY('82 Devonshire Street', 'Boston', '02109')
JMVYX95	JPMorgan	ADDRESS_TY('1111 Polaris Parkway', 'Columbus', '43240')
VFIAX62	Vanguard	ADDRESS_TY('PO Box 2600', 'Valley Forge', '19482')



FFAMILYID	MFUNDNAME	FAMILYADDRESS(STREETNAME, CITYNAME, ZIPCODE)
HLIEX25	JPMorgan	ADDRESS_TY('245 Park Avenue', 'New York', '10167')
RPMGX44	T. Rowe Price	ADDRESS_TY('100 East Pratt Street', 'Baltimore', '21202')
FXAIX01	Fidelity Investments	ADDRESS_TY('82 Devonshire Street', 'Boston', '02109')
JMVYX95	JPMorgan	ADDRESS_TY('1111 Polaris Parkway', 'Columbus', '43240')
VFIAX62	Vanguard	ADDRESS_TY('PO Box 2600', 'Valley Forge', '19482')

6. Mutual_Fund Table

Database creation script

```
--Create Table Mutual_Fund
CREATE TABLE Mutual_Fund
(fundTicker VARCHAR2(10) NOT NULL,
mFundName VARCHAR2(40) NOT NULL,
prinObjective VARCHAR2(20),
mFundCurrentPrice NUMBER(10,2),
mFundAnnualHigh NUMBER(10,2),
mFundAnnualLow NUMBER(10,2),
percentYield NUMBER(10,2),
mFundFamilyID VARCHAR2(10),
CONSTRAINT mutual_fund_fundTicker_pk PRIMARY KEY (fundTicker));
```

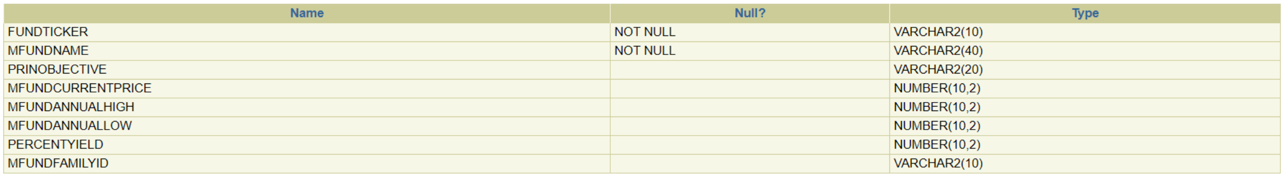
Add Foreign Keys

```
--Create Foreign Key in Mutual_Fund
ALTER TABLE Mutual_Fund
ADD FOREIGN KEY (mFundFamilyID) REFERENCES Fund_Family(fFamilyID);
```

Database structure/relational schema

```
--Describe Table Mutual_Fund
DESC Mutual_Fund;
```

Name	Null?	Type
FUNDTICKER	NOT NULL	VARCHAR2(10)
MFUNDNAME	NOT NULL	VARCHAR2(40)
PRINOBJECTIVE		VARCHAR2(20)
MFUNDCURRENTPRICE		NUMBER(10,2)
MFUNDANNUALHIGH		NUMBER(10,2)
MFUNDANNUALLOW		NUMBER(10,2)
PERCENTYIELD		NUMBER(10,2)
MFUNDFAMILYID		VARCHAR2(10)



Name	Null?	Type
FUNDTICKER	NOT NULL	VARCHAR2(10)
MFUNDNAME	NOT NULL	VARCHAR2(40)
PRINOBJECTIVE		VARCHAR2(20)
MFUNDCURRENTPRICE		NUMBER(10,2)
MFUNDANNUALHIGH		NUMBER(10,2)
MFUNDANNUALLOW		NUMBER(10,2)
PERCENTYIELD		NUMBER(10,2)
MFUNDFAMILYID		VARCHAR2(10)

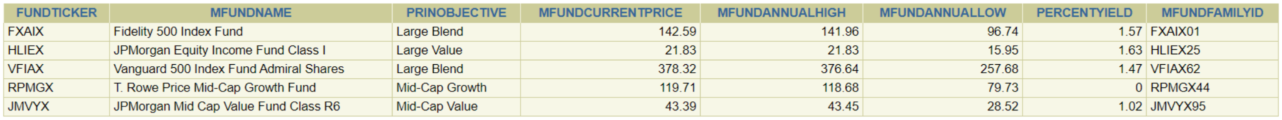
Database insert statement

```
--Insert into Table Mutual_Fund
INSERT INTO Mutual_Fund values
('FXAIX','Fidelity 500 Index Fund','Large Blend', 142.59, 141.96, 96.74, 1.57, 'FXAIX01');
INSERT INTO Mutual_Fund values
('HLIEX','JPMorgan Equity Income Fund Class I','Large Value', 21.83, 21.83, 15.95, 1.63, 'HLIEX25');
INSERT INTO Mutual_Fund values
('VFIAX','Vanguard 500 Index Fund Admiral Shares','Large Blend', 378.32, 376.64, 257.68, 1.47,
'VFIAX62');
INSERT INTO Mutual_Fund values
('RPMGX','T. Rowe Price Mid-Cap Growth Fund','Mid-Cap Growth', 119.71, 118.68, 79.73, 0.00,
'RPMGX44');
INSERT INTO Mutual_Fund values
('JMVYX','JPMorgan Mid Cap Value Fund Class R6','Mid-Cap Value', 43.39, 43.45, 28.52, 1.02,
'JMVYX95');
```

Database instance

```
--Select * from Table Mutual_Fund
SELECT *
FROM Mutual_Fund;
```

FUNDTICKER	MFUNDNAME	PRINOBJECTIVE	MFUNDCURRENTPRICE	MFUNDANNUALHIGH	MFUNDANNUALLOW	PERCENTYIELD	MFUNDFAMILYID
FXAIX	Fidelity 500 Index Fund	Large Blend	142.59	141.96	96.74	1.57	FXAIX01
HLIEX	JPMorgan Equity Income Fund Class I	Large Value	21.83	21.83	15.95	1.63	HLIEX25
VFIAX	Vanguard 500 Index Fund Admiral Shares	Large Blend	378.32	376.64	257.68	1.47	VFIAX62
RPMGX	T. Rowe Price Mid-Cap Growth Fund	Mid-Cap Growth	119.71	118.68	79.73	0	RPMGX44
JMVYX	JPMorgan Mid Cap Value Fund Class R6	Mid-Cap Value	43.39	43.45	28.52	1.02	JMVYX95



FUNDTICKER	MFUNDNAME	PRINOBJECTIVE	MFUNDCURRENTPRICE	MFUNDANNUALHIGH	MFUNDANNUALLOW	PERCENTYIELD	MFUNDFAMILYID
FXAIX	Fidelity 500 Index Fund	Large Blend	142.59	141.96	96.74	1.57	FXAIX01
HLIEX	JPMorgan Equity Income Fund Class I	Large Value	21.83	21.83	15.95	1.63	HLIEX25
VFIAX	Vanguard 500 Index Fund Admiral Shares	Large Blend	378.32	376.64	257.68	1.47	VFIAX62
RPMGX	T. Rowe Price Mid-Cap Growth Fund	Mid-Cap Growth	119.71	118.68	79.73	0	RPMGX44
JMVYX	JPMorgan Mid Cap Value Fund Class R6	Mid-Cap Value	43.39	43.45	28.52	1.02	JMVYX95

7. Stock_Portfolio Table

Database creation script

```
--Create Table Stock_Portfolio
CREATE TABLE Stock_Portfolio
(taxID VARCHAR2 (10) NOT NULL,
 stockTicker VARCHAR2(5) NOT NULL,
 sNumberOfShares NUMBER(6),
CONSTRAINT stock_portfolio_ck_pk PRIMARY KEY (taxID, stockTicker));
```

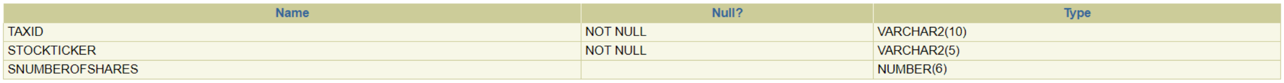
Add Foreign Keys

```
--Add Foreign Keys to Stock_Portfolio
ALTER TABLE Stock_Portfolio
ADD FOREIGN KEY (taxID) REFERENCES Client(taxID);
ALTER TABLE Stock_Portfolio
ADD FOREIGN KEY (stockTicker) REFERENCES Stock(stockTicker);
```

Database structure/relational schema

```
--Describe Table Stock_Portfolio
DESC Stock_Portfolio;
```

Name	Null?	Type
TAXID	NOT NULL	VARCHAR2(10)
STOCKTICKER	NOT NULL	VARCHAR2(5)
SNUMBEROFSHARES		NUMBER(6)



Name	Null?	Type
TAXID	NOT NULL	VARCHAR2(10)
STOCKTICKER	NOT NULL	VARCHAR2(5)
SNUMBEROFSHARES		NUMBER(6)

Database insert statement

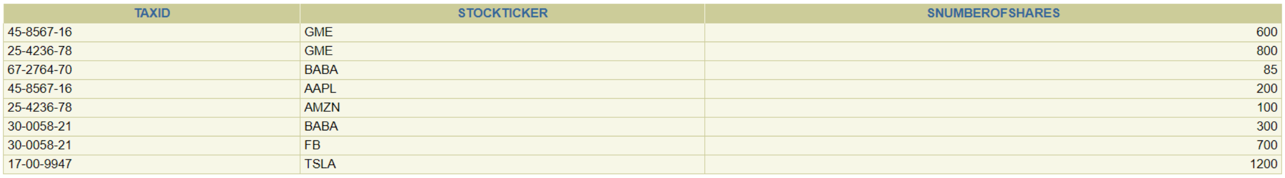
```
--Insert into Table Stock_Portfolio
INSERT INTO Stock_Portfolio values
('45-8567-16','GME','600');
INSERT INTO Stock_Portfolio values
('25-4236-78','GME','800');
INSERT INTO Stock_Portfolio values
('67-2764-70','BABA','85');
INSERT INTO Stock_Portfolio values
('45-8567-16','AAPL','200');
INSERT INTO Stock_Portfolio values
```

```
('25-4236-78','AMZN','100');
INSERT INTO Stock_Portfolio values
('30-0058-21','BABA','300');
INSERT INTO Stock_Portfolio values
('30-0058-21','FB','700');
INSERT INTO Stock_Portfolio values
('17-00-9947','TSLA','1200');
```

Database instance

```
--Select * from Table Stock_Portfolio
SELECT *
FROM Stock_Portfolio;
```

TAXID	STOCKTICKER	SNUMBEROFSHARES
45-8567-16	GME	600
25-4236-78	GME	800
67-2764-70	BABA	85
45-8567-16	AAPL	200
25-4236-78	AMZN	100
30-0058-21	BABA	300
30-0058-21	FB	700
17-00-9947	TSLA	1200



TAXID	STOCKTICKER	SNUMBEROFSHARES
45-8567-16	GME	600
25-4236-78	GME	800
67-2764-70	BABA	85
45-8567-16	AAPL	200
25-4236-78	AMZN	100
30-0058-21	BABA	300
30-0058-21	FB	700
17-00-9947	TSLA	1200

8. Mutual_Fund_Portfolio Table

Database creation script

```
--Create Table Mutual_Fund_Portfolio
CREATE TABLE Mutual_Fund_Portfolio
(taxID VARCHAR2 (10) NOT NULL,
 fundTicker VARCHAR2(5) NOT NULL,
 mFundNumberOfShares NUMBER(6),
CONSTRAINT mutual_fund_portfolio_ck_pk PRIMARY KEY (taxID, fundTicker));
```

Add Foreign Keys

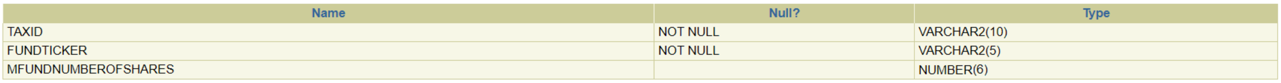
```
--Add Foreign Keys to Mutual_Fund_Portfolio
ALTER TABLE Mutual_Fund_Portfolio
```

```
ADD FOREIGN KEY (taxID) REFERENCES Client(taxID);
ALTER TABLE Mutual_Fund_Portfolio
ADD FOREIGN KEY (fundTicker) REFERENCES Mutual_Fund(fundTicker);
```

Database structure/relational schema

```
--Describe Table Mutual_Fund_Portfolio
DESC Mutual_Fund_Portfolio;
```

Name	Null?	Type
TAXID	NOT NULL	VARCHAR2(10)
FUNDTICKER	NOT NULL	VARCHAR2(5)
MFUNDNUMBEROFSHARES		NUMBER(6)



Name	Null?	Type
TAXID	NOT NULL	VARCHAR2(10)
FUNDTICKER	NOT NULL	VARCHAR2(5)
MFUNDNUMBEROFSHARES		NUMBER(6)

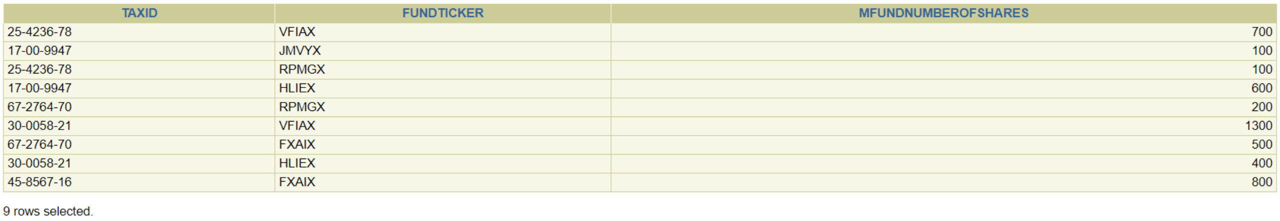
Database insert statement

```
--Insert into Table Mutual_Fund_Portfolio
Insert Into Mutual_Fund_Portfolio values
('25-4236-78','VFIAX','700');
Insert Into Mutual_Fund_Portfolio values
('17-00-9947','JMVYX','100');
Insert Into Mutual_Fund_Portfolio values
('25-4236-78','RPMGX','100');
Insert Into Mutual_Fund_Portfolio values
('17-00-9947','HLIEX','600');
Insert Into Mutual_Fund_Portfolio values
('67-2764-70','RPMGX','200');
Insert Into Mutual_Fund_Portfolio values
('30-0058-21','VFIAX','1300');
Insert Into Mutual_Fund_Portfolio values
('67-2764-70','FXAIX','500');
Insert Into Mutual_Fund_Portfolio values
('30-0058-21','HLIEX','400');
Insert Into Mutual_Fund_Portfolio values
('45-8567-16','FXAIX','800');
```

Database instance

```
--Select * from Table Mutual_Fund_Portfolio
SELECT *
FROM Mutual_Fund_Portfolio;
```

TAXID	FUNDTICKER	MFUNDNUMBEROFSHARES
25-4236-78	VFIAX	700
17-00-9947	JMVYX	100
25-4236-78	RPMGX	100
17-00-9947	HLIEX	600
67-2764-70	RPMGX	200
30-0058-21	VFIAX	1300
67-2764-70	FXAIX	500
30-0058-21	HLIEX	400
45-8567-16	FXAIX	800



TAXID	FUNDTICKER	MFUNDNUMBEROFSHARES
25-4236-78	VFIAX	700
17-00-9947	JMVYX	100
25-4236-78	RPMGX	100
17-00-9947	HLIEX	600
67-2764-70	RPMGX	200
30-0058-21	VFIAX	1300
67-2764-70	FXAIX	500
30-0058-21	HLIEX	400
45-8567-16	FXAIX	800

9 rows selected.

```
COMMIT;
```

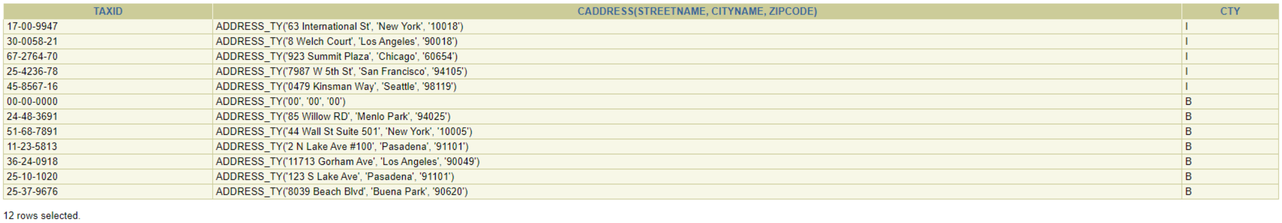
6. List ONE Insert, Update, Delete and CREATE View in each Group

<u>Insert</u> : [Ninelia]

```
--Insert a row into Table Client
INSERT INTO Client values
('25-37-9676', address_ty('8039 Beach Blvd', 'Buena Park', '90620'), 'B');
```

```
--Select * from Table Client
SELECT *
FROM Client;
```

TAXID	CADDRESS(STREETNAME, CITYNAME, ZIPCODE)	CTY
17-00-9947	ADDRESS_TY('63 International St', 'New York', '10018')	I
30-0058-21	ADDRESS_TY('8 Welch Court', 'Los Angeles', '90018')	I
67-2764-70	ADDRESS_TY('923 Summit Plaza', 'Chicago', '60654')	I
25-4236-78	ADDRESS_TY('7987 W 5th St', 'San Francisco', '94105')	I
45-8567-16	ADDRESS_TY('0479 Kinsman Way', 'Seattle', '98119')	I
00-00-0000	ADDRESS_TY('00', '00', '00')	B
24-48-3691	ADDRESS_TY('85 Willow RD', 'Menlo Park', '94025')	B
51-68-7891	ADDRESS_TY('44 Wall St Suite 501', 'New York', '10005')	B
11-23-5813	ADDRESS_TY('2 N Lake Ave #100', 'Pasadena', '91101')	B
36-24-0918	ADDRESS_TY('11713 Gorham Ave', 'Los Angeles', '90049')	B
25-10-1020	ADDRESS_TY('123 S Lake Ave', 'Pasadena', '91101')	B
25-37-9676	ADDRESS_TY('8039 Beach Blvd', 'Buena Park', '90620')	B



TAXID	CADDRESS(STREETNAME, CITYNAME, ZIPCODE)	CTY
17-00-9947	ADDRESS_TY('63 International St', 'New York', '10018')	I
30-0058-21	ADDRESS_TY('8 Welch Court', 'Los Angeles', '90018')	I
67-2764-70	ADDRESS_TY('923 Summit Plaza', 'Chicago', '60654')	I
25-4236-78	ADDRESS_TY('7987 W 5th St', 'San Francisco', '94105')	I
45-8567-16	ADDRESS_TY('0479 Kinsman Way', 'Seattle', '98119')	I
00-00-0000	ADDRESS_TY('00', '00', '00')	B
24-48-3691	ADDRESS_TY('85 Willow RD', 'Menlo Park', '94025')	B
51-68-7891	ADDRESS_TY('44 Wall St Suite 501', 'New York', '10005')	B
11-23-5813	ADDRESS_TY('2 N Lake Ave #100', 'Pasadena', '91101')	B
36-24-0918	ADDRESS_TY('11713 Gorham Ave', 'Los Angeles', '90049')	B
25-10-1020	ADDRESS_TY('123 S Lake Ave', 'Pasadena', '91101')	B
25-37-9676	ADDRESS_TY('8039 Beach Blvd', 'Buena Park', '90620')	B

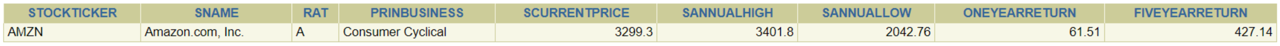
12 rows selected.

<u>Update</u> : [Savita]

```
--Update a row in Table Stock
UPDATE Stock
SET ratings = 'A'
WHERE stockTicker = 'AMZN';
```

```
--Select specific row from Table Stock
SELECT *
FROM Stock
WHERE stockTicker = 'AMZN';
```

STOCKTICKER	SNAME	RAT	PRINBUSINESS	SCURRENTPRICE	SANNUALHIGH	SANNUALLOW	ONEYEARRETURN	FIVEYEARRETURN
AMZN	Amazon.com, Inc.	A	Consumer Cyclical	3299.3	3401.8	2042.76	61.51	427.14

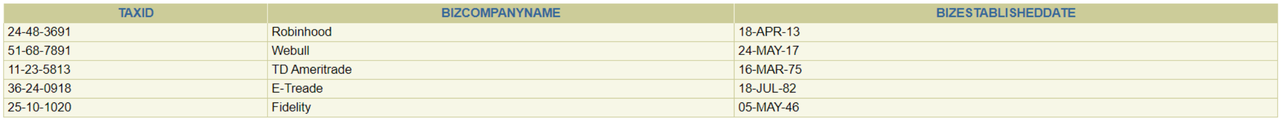


STOCKTICKER	SNAME	RAT	PRINBUSINESS	SCURRENTPRICE	SANNUALHIGH	SANNUALLOW	ONEYEARRETURN	FIVEYEARRETURN
AMZN	Amazon.com, Inc.	A	Consumer Cyclical	3299.3	3401.8	2042.76	61.51	427.14

<u>Delete</u> : [Davide]

```
--Delete a row from Table Business
DELETE
FROM Business
WHERE taxID = '00-00-0000';

--Select * from Table Business
SELECT *
FROM Business;
```

TAXID	BIZCOMPANYNAME	BIZESTABLISHEDDATE
24-48-3691	Robinhood	18-APR-13
51-68-7891	Webull	24-MAY-17
11-23-5813	TD Ameritrade	16-MAR-75
36-24-0918	E-Treade	18-JUL-82
25-10-1020	Fidelity	05-MAY-46

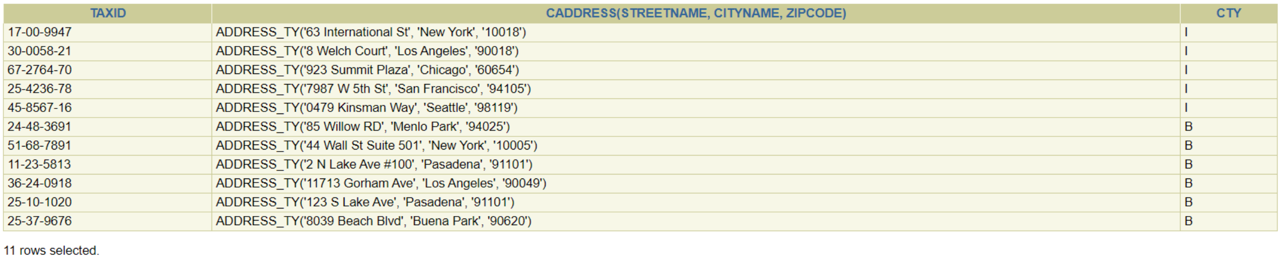


TAXID	BIZCOMPANYNAME	BIZESTABLISHEDDATE
24-48-3691	Robinhood	18-APR-13
51-68-7891	Webull	24-MAY-17
11-23-5813	TD Ameritrade	16-MAR-75
36-24-0918	E-Treade	18-JUL-82
25-10-1020	Fidelity	05-MAY-46

```
--Delete a row from Table Client
DELETE
FROM Client
WHERE taxID = '00-00-0000';

--Select * from Table Client
SELECT *
FROM Client;
```

TAXID	CADDRESS(STREETNAME, CITYNAME, ZIPCODE)	CTY
17-00-9947	ADDRESS_TY('63 International St', 'New York', '10018')	I
30-0058-21	ADDRESS_TY('8 Welch Court', 'Los Angeles', '90018')	I
67-2764-70	ADDRESS_TY('923 Summit Plaza', 'Chicago', '60654')	I
25-4236-78	ADDRESS_TY('7987 W 5th St', 'San Francisco', '94105')	I
45-8567-16	ADDRESS_TY('0479 Kinsman Way', 'Seattle', '98119')	I
24-48-3691	ADDRESS_TY('85 Willow RD', 'Menlo Park', '94025')	B
51-68-7891	ADDRESS_TY('44 Wall St Suite 501', 'New York', '10005')	B
11-23-5813	ADDRESS_TY('2 N Lake Ave #100', 'Pasadena', '91101')	B
36-24-0918	ADDRESS_TY('11713 Gorham Ave', 'Los Angeles', '90049')	B

25-10-1020	ADDRESS_TY('123 S Lake Ave', 'Pasadena', '91101')	B
25-37-9676	ADDRESS_TY('8039 Beach Blvd', 'Buena Park', '90620')	B



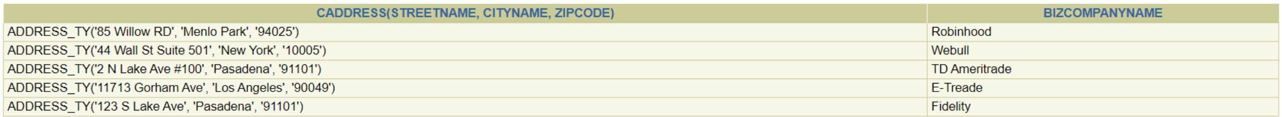
TAXID	CADDRESS(STREETNAME, CITYNAME, ZIPCODE)	CTY
17-00-9947	ADDRESS_TY('63 International St', 'New York', '10018')	I
30-0058-21	ADDRESS_TY('8 Welch Court', 'Los Angeles', '90018')	I
67-2764-70	ADDRESS_TY('923 Summit Plaza', 'Chicago', '60654')	I
25-4236-78	ADDRESS_TY('7987 W 5th St', 'San Francisco', '94105')	I
45-8567-16	ADDRESS_TY('0479 Kinsman Way', 'Seattle', '98119')	I
24-48-3691	ADDRESS_TY('85 Willow RD', 'Menlo Park', '94025')	B
51-68-7891	ADDRESS_TY('44 Wall St Suite 501', 'New York', '10005')	B
11-23-5813	ADDRESS_TY('2 N Lake Ave #100', 'Pasadena', '91101')	B
36-24-0918	ADDRESS_TY('11713 Gorham Ave', 'Los Angeles', '90049')	B
25-10-1020	ADDRESS_TY('123 S Lake Ave', 'Pasadena', '91101')	B
25-37-9676	ADDRESS_TY('8039 Beach Blvd', 'Buena Park', '90620')	B

11 rows selected.

View : [Raymond]

```
--Create a View for Business Clients
CREATE VIEW Business_Clients As
SELECT Client.cAddress,  Business.bizCompanyName
FROM Client
INNER JOIN Business ON Client.taxID=Business.taxID;

--To Test our view
SELECT *
FROM Business_Clients;
```

CADDRESS(STREETNAME, CITYNAME, ZIPCODE)	BIZCOMPANYNAME
ADDRESS_TY('2 N Lake Ave #100', 'Pasadena', '91101')	TD Ameritrade
ADDRESS_TY('85 Willow RD', 'Menlo Park', '94025')	Robinhood
ADDRESS_TY('123 S Lake Ave', 'Pasadena', '91101')	Fidelity
ADDRESS_TY('11713 Gorham Ave', 'Los Angeles', '90049')	E-Treade
ADDRESS_TY('44 Wall St Suite 501', 'New York', '10005')	Webull



CADDRESS(STREETNAME, CITYNAME, ZIPCODE)	BIZCOMPANYNAME
ADDRESS_TY('85 Willow RD', 'Menlo Park', '94025')	Robinhood
ADDRESS_TY('44 Wall St Suite 501', 'New York', '10005')	Webull
ADDRESS_TY('2 N Lake Ave #100', 'Pasadena', '91101')	TD Ameritrade
ADDRESS_TY('11713 Gorham Ave', 'Los Angeles', '90049')	E-Treade
ADDRESS_TY('123 S Lake Ave', 'Pasadena', '91101')	Fidelity

7. Testing the database with query questions and showing SQL solutions and output

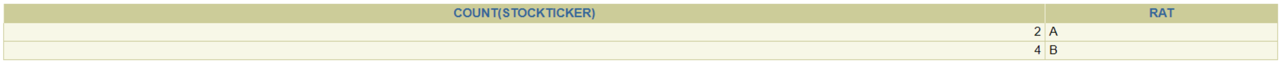
Test your relational database with SELECT Statements via ORACLE SQL*PLUS, which include SQL queries, SQL solutions and corresponding output and save them in the MS word file. One member writes one SELECT…. Your SELECT statements must include join tables, subqueries, Group by …. Having and function statements.

Query 1: [Davide]

List the number of stocks based on stock rating.

```
SELECT Count(stockTicker), ratings
FROM Stock
GROUP BY ratings;
```

COUNT(STOCKTICKER)	RAT
2	A
4	B



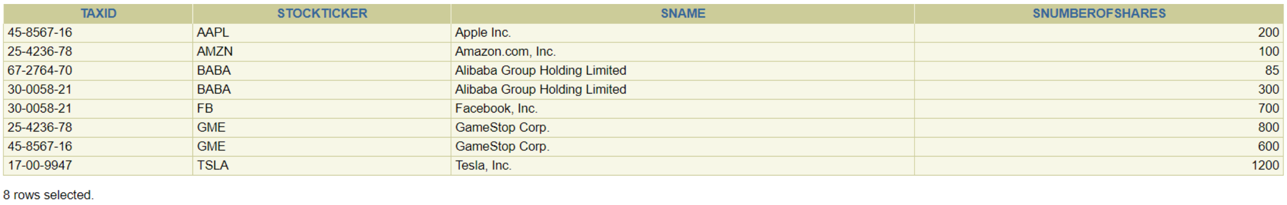
COUNT(STOCKTICKER)	RAT
2	A
4	B

Query 2: [Savita]

List the client's tax id, stock ticker, company name, and number of shares by the stock ticker in ascending order.

```
SELECT Client.taxID, Stock_Portfolio.stockTicker, Stock.sname, Stock_Portfolio.sNumberOfShares
FROM Stock_Portfolio
LEFT JOIN Client ON  Client.taxID = Stock_Portfolio.taxID
LEFT JOIN Stock ON Stock.stockTicker = Stock_Portfolio.stockTicker
ORDER BY Stock_Portfolio.stockTicker;
```

TAXID	STOCKTICKER	SNAME	SNUMBEROFSHARES
45-8567-16	AAPL	Apple Inc.	200
25-4236-78	AMZN	Amazon.com, Inc.	100
67-2764-70	BABA	Alibaba Group Holding Limited	85
30-0058-21	BABA	Alibaba Group Holding Limited	300
30-0058-21	FB	Facebook, Inc.	700
25-4236-78	GME	GameStop Corp.	800
45-8567-16	GME	GameStop Corp.	600
17-00-9947	TSLA	Tesla, Inc.	1200



TAXID	STOCKTICKER	SNAME	SNUMBEROFSHARES
45-8567-16	AAPL	Apple Inc.	200
25-4236-78	AMZN	Amazon.com, Inc.	100
67-2764-70	BABA	Alibaba Group Holding Limited	85
30-0058-21	BABA	Alibaba Group Holding Limited	300
30-0058-21	FB	Facebook, Inc.	700
25-4236-78	GME	GameStop Corp.	800
45-8567-16	GME	GameStop Corp.	600
17-00-9947	TSLA	Tesla, Inc.	1200

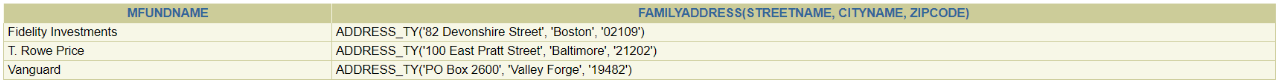
8 rows selected.

Query 3: [Raymond]

List the name of the mutual fund and the address of its corresponding fund family, where the annual high for the fund family is greater than $100.

```
SELECT mFundName, familyAddress
FROM Fund_Family
WHERE fFamilyID IN(
    SELECT MFUNDFAMILYID
    FROM Mutual_Fund
    WHERE mFundAnnualHigh > 100);
```

MFUNDNAME	FAMILYADDRESS(STREETNAME, CITYNAME, ZIPCODE)
Fidelity Investments	ADDRESS_TY('82 Devonshire Street', 'Boston', '02109')
T. Rowe Price	ADDRESS_TY('100 East Pratt Street', 'Baltimore', '21202')
Vanguard	ADDRESS_TY('PO Box 2600', 'Valley Forge', '19482')



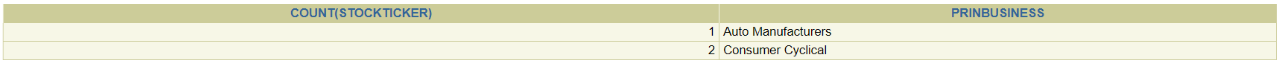
MFUNDNAME	FAMILYADDRESS(STREETNAME, CITYNAME, ZIPCODE)
Fidelity Investments	ADDRESS_TY('82 Devonshire Street', 'Boston', '02109')
T. Rowe Price	ADDRESS_TY('100 East Pratt Street', 'Baltimore', '21202')
Vanguard	ADDRESS_TY('PO Box 2600', 'Valley Forge', '19482')

Query 4: [Ninelia]

List the industries of stocks that have an average current price which is greater than 500.

```
SELECT Count(stockTicker), prinBusiness
FROM Stock
GROUP BY prinBusiness
HAVING AVG(sCurrentPrice) > 500;
```

COUNT(STOCKTICKER)	PRINBUSINESS
1	Auto Manufacturers
2	Consumer Cyclical



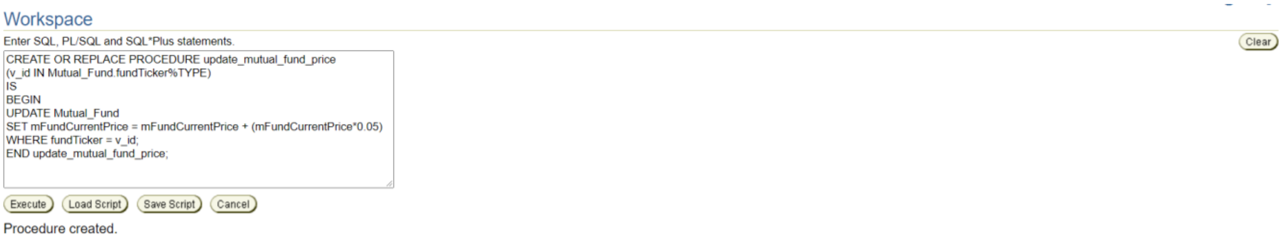
COUNT(STOCKTICKER)	PRINBUSINESS
1	Auto Manufacturers
2	Consumer Cyclical

8. PL/SQL Statement blocks

PL/SQL 1: [Davide]

--The PL/SQL script to create a function to do a 5% increase in mutual fund price.

```
CREATE OR REPLACE PROCEDURE update_mutual_fund_price
(v_id IN Mutual_Fund.fundTicker%TYPE)
IS
BEGIN
UPDATE Mutual_Fund
SET mFundCurrentPrice = mFundCurrentPrice + (mFundCurrentPrice*0.05)
WHERE fundTicker = v_id;
END update_mutual_fund_price;
```

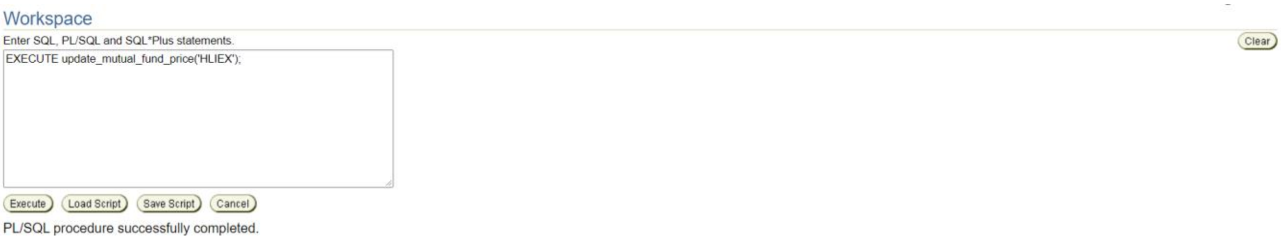


```
CREATE OR REPLACE PROCEDURE update_mutual_fund_price
(v_id IN Mutual_Fund.fundTicker%TYPE)
IS
BEGIN
UPDATE Mutual_Fund
SET mFundCurrentPrice = mFundCurrentPrice + (mFundCurrentPrice*0.05)
WHERE fundTicker = v_id;
END update_mutual_fund_price;
```

```
Procedure created.
```

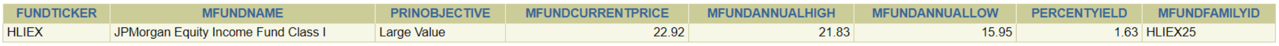
--The PL/SQL Script to test the procedure

```
EXECUTE update_mutual_fund_price ('HLIEX');
```



```
EXECUTE update_mutual_fund_price('HLIEX');
```

```
PL/SQL procedure successfully completed.
```

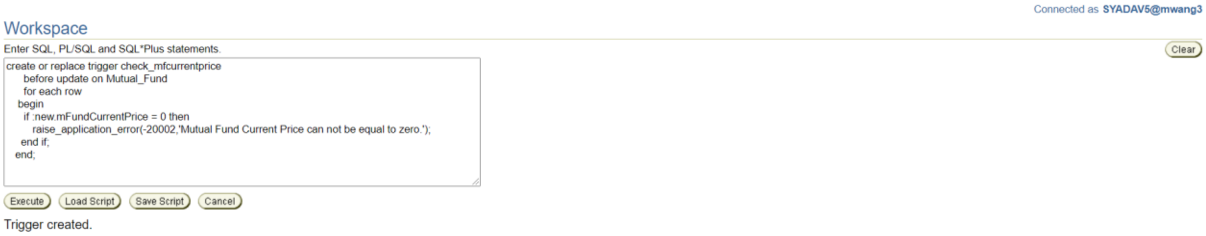


FUNDTICKER	MFUNDNAME	PRINOBJECTIVE	MFUNDCURRENTPRICE	MFUNDANNUALHIGH	MFUNDANNUALLOW	PERCENTYIELD	MFUNDFAMILYID
HLIEX	JPMorgan Equity Income Fund Class I	Large Value	22.92	21.83	15.95	1.63	HLIEX25

PL/SQL 2: [Savita]

--The PL/SQL script creates a Trigger to check the current mutual fund price is zero and raise an application error.

```
CREATE OR REPLACE TRIGGER check_mfcurrentprice
      before update on Mutual_Fund
      for each row
    BEGIN
      if :new.mFundCurrentPrice = 0 then
         raise_application_error(-20002,'Mutual Fund Current Price cannot be equal to zero.');
     end if;
   END;
```

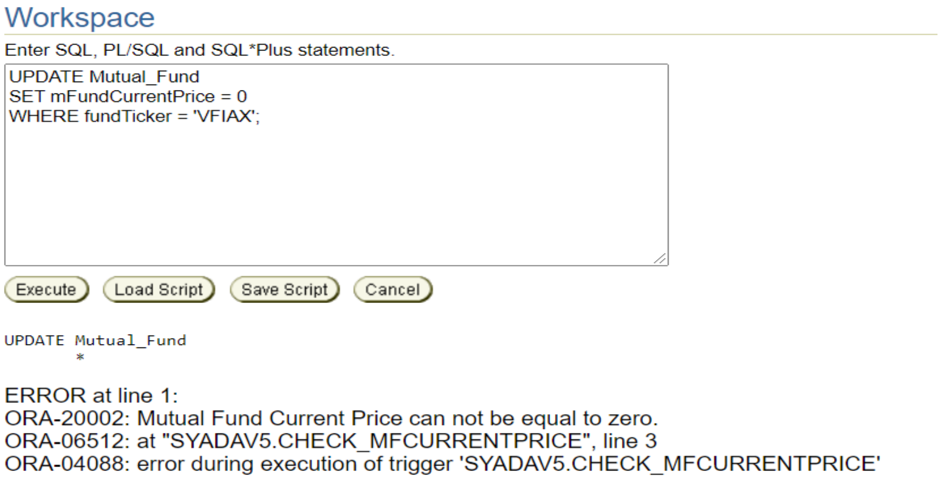

Connected as SYADAV5@mwang3
Workspace
Enter SQL, PL/SQL and SQL*Plus statements.
Clear

```
create or replace trigger check_mfcurrentprice
   before update on Mutual_Fund
   for each row
  begin
   if :new.mFundCurrentPrice = 0 then
     raise_application_error(-20002,'Mutual Fund Current Price can not be equal to zero.');
  end if;
 end;
```

Execute
Load Script
Save Script
Cancel

```
Trigger created.
```

```
--The PL/Script to test the trigger
UPDATE Mutual_Fund
SET mFundCurrentPrice = 0
WHERE fundTicker = 'VFIAX';
```



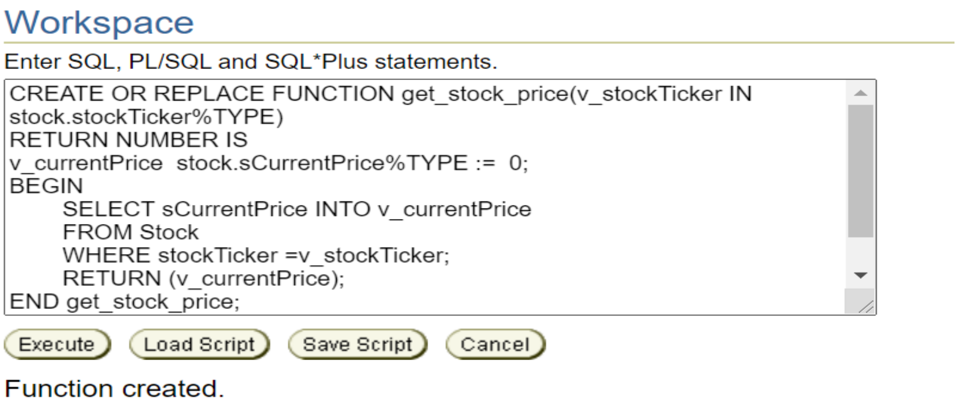
```
UPDATE Mutual_Fund
SET mFundCurrentPrice = 0
WHERE fundTicker = 'VFIAX';
```

```
UPDATE Mutual_Fund
       *

ERROR at line 1:
ORA-20002: Mutual Fund Current Price can not be equal to zero.
ORA-06512: at "SYADAV5.CHECK_MFCURRENTPRICE", line 3
ORA-04088: error during execution of trigger 'SYADAV5.CHECK_MFCURRENTPRICE'
```

<u>PL/SQL 3</u>: [Raymond]

--The PL/SQL script to create a function to return the current stock price that corresponds to the stock ticker.

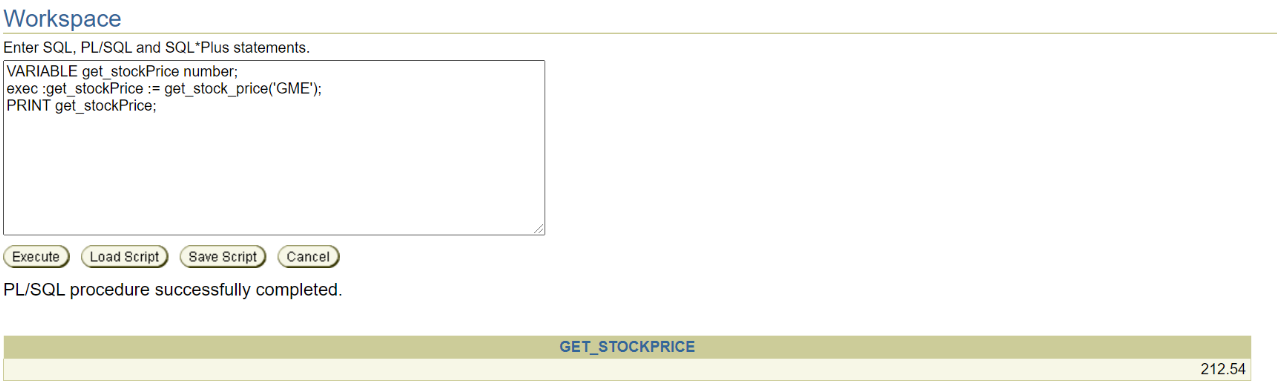
```
CREATE OR REPLACE FUNCTION get_stock_price(v_stockTicker IN stock.stockTicker%TYPE)
RETURN NUMBER IS
v_currentPrice  stock.sCurrentPrice%TYPE :=  0;
BEGIN
          SELECT sCurrentPrice INTO v_currentPrice
          FROM Stock
          WHERE stockTicker =v_stockTicker;
          RETURN (v_currentPrice);
END get_stock_price;
```



```
CREATE OR REPLACE FUNCTION get_stock_price(v_stockTicker IN
stock.stockTicker%TYPE)
RETURN NUMBER IS
v_currentPrice  stock.sCurrentPrice%TYPE :=  0;
BEGIN
        SELECT sCurrentPrice INTO v_currentPrice
        FROM Stock
        WHERE stockTicker =v_stockTicker;
        RETURN (v_currentPrice);
END get_stock_price;
```

```
Function created.
```

```
--The PL/SQL Script to test the function
VARIABLE get_stockPrice number;
exec :get_stockPrice := get_stock_price('GME');
PRINT get_stockPrice;
```



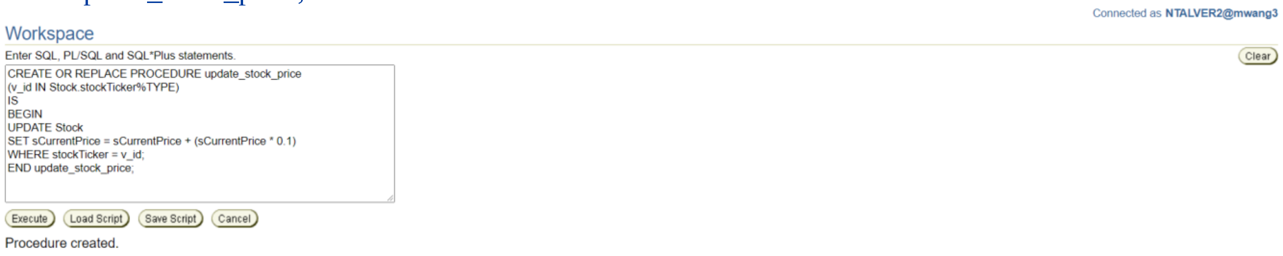
```
VARIABLE get_stockPrice number;
exec :get_stockPrice := get_stock_price('GME');
PRINT get_stockPrice;
```

```
PL/SQL procedure successfully completed.
```

GET_STOCKPRICE
212.54

PL/SQL 4: [Ninelia]

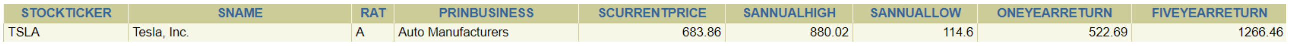
```
--The PL/SQL script to create a function to increase the stock price by 10% for a record.

CREATE OR REPLACE PROCEDURE update_stock_price
(v_id IN Stock.stockTicker%TYPE)
IS
BEGIN
UPDATE Stock
SET sCurrentPrice = sCurrentPrice + (sCurrentPrice * 0.1)
WHERE stockTicker = v_id;
END update_stock_price;
```


Connected as NTALVER2@mwang3
Workspace
Enter SQL, PL/SQL and SQL*Plus statements.
Clear
Execute
Load Script
Save Script
Cancel

```
CREATE OR REPLACE PROCEDURE update_stock_price
(v_id IN Stock.stockTicker%TYPE)
IS
BEGIN
UPDATE Stock
SET sCurrentPrice = sCurrentPrice + (sCurrentPrice * 0.1)
WHERE stockTicker = v_id;
END update_stock_price;
```

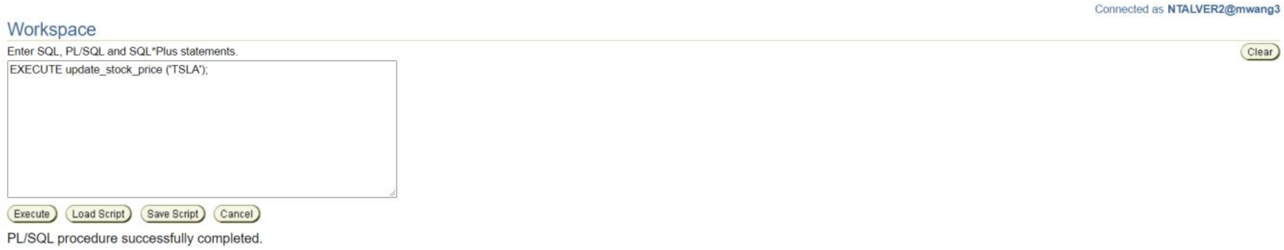
```
Procedure created.
```

```
--Select * from Table Stock (BEFORE)
SELECT *
FROM Stock
WHERE stockticker = 'TSLA';
```



STOCKTICKER	SNAME	RAT	PRINBUSINESS	SCURRENTPRICE	SANNUALHIGH	SANNUALLOW	ONEYEARRETURN	FIVEYEARRETURN
TSLA	Tesla, Inc.	A	Auto Manufacturers	683.86	880.02	114.6	522.69	1266.46

```
--The PL/SQL Script to test the procedure
EXECUTE update_stock_price ('TSLA');
```



Connected as NTALVER2@mwang3

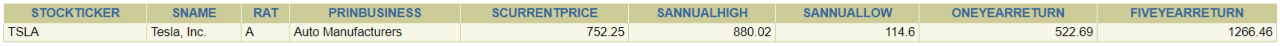
Workspace

Enter SQL, PL/SQL and SQL*Plus statements.

```
EXECUTE update_stock_price ('TSLA');
```

PL/SQL procedure successfully completed.

```
--Select * from Table Stock (AFTER)
SELECT *
FROM Stock
WHERE stockticker = 'TSLA';
```



STOCKTICKER	SNAME	RAT	PRINBUSINESS	SCURRENTPRICE	SANNUALHIGH	SANNUALLOW	ONEYEARRETURN	FIVEYEARRETURN
TSLA	Tesla, Inc.	A	Auto Manufacturers	752.25	880.02	114.6	522.69	1266.46

9. Include Object Types in the object column or object tables

Object 1

```
--Create Object Date Of Birth and Gender in Individual Clients
CREATE OR REPLACE TYPE personal_info_ty as OBJECT
(indvDOB DATE,
 indvGender VARCHAR2(1));
```

Object 2

```
--Create Object Address
CREATE OR REPLACE TYPE address_ty AS OBJECT
(streetName VARCHAR2(50),
 cityName VARCHAR2(20),
 zipCode VARCHAR2(5));
```

Object 3

```
--Create Object Name in Individual Clients
CREATE OR REPLACE TYPE name_ty AS OBJECT
(indvFirstName VARCHAR2(20),
 indvLastName VARCHAR2(20));
```