

Oracle Database Design and Implementation

using SQL Developer and iSQL Plus

Gregory Mamoyac, Nikita Shendkar, Ruchita Shinde, Nitesh Kamboj

California State University, Los Angeles

Author Note

The Research Group is now at the College of Business and Economics, California State University, Los Angeles. This research is supported by CIS 5430 Database Systems led by Dr. Ming Wang.

Correspondence concerning this article should be addressed the Research Group, M.S. Information Systems Program, College of Business and Economics, California State University, Los Angeles, 5151 State University Dr., Los Angeles, CA 90032.

Table of Contents

Oracle Database Design and Implementation – 4

Entity Types and Relationship Matrix – 4

Generated Business Rules – 4

ER Diagram – 5

Database Logical Design – 6

Referential Integrity – 7

Normalization – 7

Normalization Table – 7

Scripts – 8

References – 35

Oracle Database Design and Implementation

This project is to develop and implement an Oracle database. See the scenario of the Investment database.

This simplified Investment Portfolio case study maintains information regarding clients that invest in stocks and mutual funds. Each client has a unique taxpayer identification number. A client may invest in stocks or mutual funds.

A stock is uniquely identified by its ticker, which is symbol on the stock exchange. The name of issue for a stock is the name of the company. A stock has a rating, which is an assessment of the stock. The principal business of a stock denotes the primary business from which the company obtains its revenue. There are several prices associated with a stock. Besides the current price of a stock, the high and low price ranges are recorded for the calendar year. The return value for a stock, which is the price appreciation, is recorded for the prior year and five-year period.

A mutual fund is also uniquely identified by a ticker symbol and has a fund name. The principal objective of the fund is also recorded (e.g. 'G' for growth, 'I' for income, and 'S' for stability). Similar to stocks, mutual funds record a current offering price and a high and low price range. A percent yield from investment income is also maintained. A mutual fund may be associated with at most one fund family.

A fund family has a unique identification number and the company's name and address. A fund family, as its name implies, can have many associated mutual funds, but must have at least one mutual fund.

Based on the high level introduction to the Investment Portfolio enterprise, assume the following:

The characteristics of a client include a unique taxpayer identification number, a name, and an address. A client can invest in many stocks in many mutual funds.

The characteristics of a stock include the stock's unique ticker on the stock exchange and additional descriptive properties that include the name of issue, a rating, the principal business, a high and low price range, the current price, and return values for the prior year and 5-year period.

The characteristics of a mutual fund include the mutual fund's unique ticker and additional descriptive properties that include the name of the fund, its principal objective, the high and low price range, the current offering price, and the fund's yield. A mutual fund may be associated with the most one fund family.

The characteristics of a fund family include a unique identification, a name, and an address. A fund family may have mutual funds and must be associated with at least one mutual fund.

Entity Types and 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

	CLIENT	STOCK	MUTUAL FUND	FAMILY FUND
CLIENT	--	INVEST	INVEST	--
STOCK	INVEST	--	--	
MUTUAL FUND	INVEST	--	--	ASSOCIATE
FAMILY FUND	--	--	ASSOCIATE	--

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.

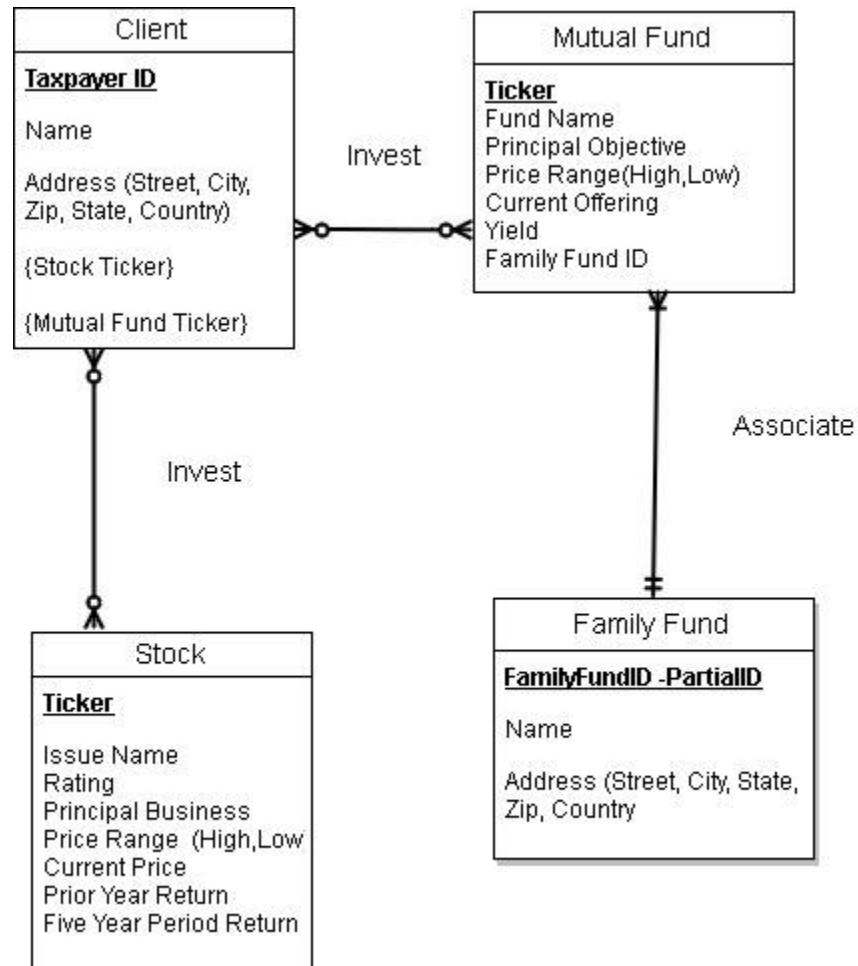
A mutual fund may be bought by one or more than one client.

One fund family must have one or more than one mutual fund.

One mutual fund must belong to only one fund family.

ER diagram

Entity types, Relationship types, Keys, Cardinality, PKs and FKs



Database Logical Design

ER diagram to a relational database schema indicating the relation name, primary key and foreign key.

NOTE: Underlined FK identifier uses referential integrity but still functions as PK in the relation

Client

<u>TaxpayerID PK</u>	FirstName	LastName	Street	City	State	Zip	Country
----------------------	-----------	----------	--------	------	-------	-----	---------

Mutual Fund

<u>Ticker PK</u>	FundName	PrincipalObjective	PriceRangeHigh	PriceRangeLow	Current Offering	Yield	FamilyFundID FK
------------------	----------	--------------------	----------------	---------------	------------------	-------	-----------------

Stock

<u>Ticker PK</u>	Issue Name	Rating	Principal Business	PriceRangeHigh	PriceRangeLow	Current Price	PriorYearReturn	FiveYearPeriodReturn
------------------	------------	--------	--------------------	----------------	---------------	---------------	-----------------	----------------------

Family Fund

<u>FamilyFundID PK</u>	<u>MutualFundTicker FK</u>	Name	Street	City	State	Zip	Country
------------------------	----------------------------	------	--------	------	-------	-----	---------

Client Stock

<u>TaxpayerID FK</u>	<u>StockTicker FK</u>	Quantity
----------------------	-----------------------	----------

Client Mutual Fund

<u>TaxpayerID FK</u>	<u>MutalFundTicker FK</u>	Quantity
----------------------	---------------------------	----------

Referential Integrity

Mutual Fund/FamilyFundID FK -> Family Fund/FamilyFundID PK

Client Stock/ TaxpayerID FK -> Client/TaxpayerID PK

Client Stock/ StockerTicker FK ->Stock/Ticker PK

Client Mutual Fund/ TaxpayerID FK -> Client/TaxpayerID PK

Client Mutual Fund/MutualFundTicker FK -> Mutual Fund/ Ticker PK

Family Fund/MutualFundTicker FK -> MutualFund/ Ticker PK

Normalization

Partial D TaxpayerID -> FirstName, LastName, Street, City, State, Zip, Country

Partial D MutualFundTicker -> FundName, PrincipalObjective, PriceRangeHigh, PriceRangeLow, CurrentOffering, Yield

Partial D StockTicker -> IssueName, Rating, PrincipalBusiness, PriceRangeHigh, PriceRangeLow, CurrentPrice, PriorYearReturn, FiveYearPeriodReturn

Partial D FamilyFundIDNumber -> Name, Street, City, State, Zip, Country

Full D TaxpayerID/StockTicker -> Quantity

Full D TaxpayerID/MutualFundTicker ->Quantity

Transitive D Rating -> PrincipalBusiness, PriceRangeHigh, PriceRangeLow, CurrentPrice, PriorYearReturn, FiveYearPeriodReturn

Transitive D Yield -> CurrentOffering

Normalization Table**NOTE:1NF Free of multivalued attributes + PK****2NF Free of Partial D + 1NF****3NF Free of Transitive D +2NF**

Table Name	1NF	2NF	3NF
Client	X	X	X
Mutual Fund	X	X	
Stock	X	X	
Family Fund	X	X	X
Client Stock	X	X	X
Client Mutual Fund	X	X	X

Script**NOTE: Script Includes DDL, DML, DCL, TCL, PLSQL****/*CREATE, ALTER, INSERT TABLES*/**

--CREATING CLIENT TABLE

```
CREATE TABLE client_table (  
TaxpayerID INT NOT NULL,  
FirstName VARCHAR (255) NOT NULL,  
LastName VARCHAR (255) NOT NULL,  
Street VARCHAR (255) NOT NULL,  
City VARCHAR (255) NOT NULL,  
State VARCHAR (255) NOT NULL,  
Zip INT NOT NULL,  
Country VARCHAR (255) NOT NULL,  
CONSTRAINT TaxpayerID_pk PRIMARY KEY (TaxpayerID)  
);
```

----INSERTING VALUES IN CLIENT TABLE

```
INSERT INTO client_table (  
TaxpayerID,  
Firstname ,  
Lastname ,  
Street ,  
City ,  
State ,  
Zip ,  
Country)  
VALUES  
(545786625,  
'Henry',  
'King',  
'1234 Fake Street',  
'Los Angeles',  
'CA',  
91754,
```



```
'USA'  
);
```

```
INSERT INTO client_table (  
TaxpayerID,  
Firstname ,  
Lastname ,  
Street ,  
City ,  
State ,  
Zip ,  
Country)  
VALUES  
(123456789,  
'Geroge',  
'Hill',  
'1324 No Street',  
'Los Angeles',  
'CA',  
91753,  
'USA'  
);
```

```
INSERT INTO client_table (  
TaxpayerID,  
Firstname ,  
Lastname ,  
Street ,  
City ,  
State ,  
Zip ,  
Country)  
VALUES  
(987654321,  
'King',  
'Pin',
```

```
'1673 Narrow Street',  
'Miami',  
'FL',  
12345,  
'USA'  
);
```

```
INSERT INTO client_table (  
TaxpayerID,  
Firstname ,  
Lastname ,  
Street ,  
City ,  
State ,  
Zip ,  
Country)  
VALUES  
(128765930,  
'Bob',  
'Marley',  
'666 High Blvd',  
'Compton',  
'CA',  
66660,  
'USA'  
);
```

```
INSERT INTO client_table (  
TaxpayerID,  
Firstname ,  
Lastname ,  
Street ,  
City ,  
State ,  
Zip ,  
Country)
```

```
VALUES
```

```
(943270000,
```

```
'Nick',
```

```
'Sanders',
```

```
'123 Green Blvd',
```

```
'Seattle',
```

```
'WA',
```

```
88765,
```

```
'USA'
```

```
);
```

```
INSERT INTO client_table (
```

```
TaxpayerID,
```

```
Firstname ,
```

```
Lastname ,
```

```
Street ,
```

```
City ,
```

```
State ,
```

```
Zip ,
```

```
Country)
```

```
VALUES
```

```
(987765401,
```

```
'Donald',
```

```
'Bautista',
```

```
'556 Tall Ave',
```

```
'Tampa',
```

```
'FL',
```

```
98987,
```

```
'USA'
```

```
);
```

```
--CREATING STOCK TABLE
```

```
CREATE TABLE stock_table (
```

```
StockID VARCHAR(255) NOT NULL,
```

```
IssueName VARCHAR (255) NOT NULL,
```

```
Rating VARCHAR (100) NOT NULL,  
PrincipalBuiness VARCHAR (255) NOT NULL,  
PriceRangeHigh Decimal(38,2) NOT NULL,  
PrinceRangeLow Decimal(38,2) NOT NULL,  
CurrentPrice Decimal(38,2) NOT NULL,  
PriorYearReturn Decimal(38,2),  
FiveYearPeriodReturn Decimal(38,2),  
CONSTRAINT StockID_pk PRIMARY KEY (StockID)  
);
```

--INSERTING VALUES IN STOCK TABLE

```
INSERT INTO stock_table  
VALUES  
(  
'12345',  
'ABC',  
'7',  
'Retail',  
123,  
100,  
102,  
109,  
108  
);
```

```
INSERT INTO stock_table  
VALUES  
(  
'67890',  
'MNO',  
'4',  
'Wholeseller',  
787,  
710,  
778,  
750,  
767  
);
```

```
INSERT INTO stock_table
VALUES
('11234',
'PQR',
'6',
'Restraunt',
500,
400,
450,
455,
480
);
```

```
INSERT INTO stock_table
VALUES
('56789',
'STU',
'2',
'Stockist',
900,
700,
800,
702,
710
);
```

```
INSERT INTO stock_table
VALUES
('21345',
'XYZ',
'1',
'Cold Storage',
1100,
1050,
1090,
```

```
1080,  
1093  
);
```

```
--CREATING CLIENTSTOCK TABLE
```

```
CREATE TABLE clientstock_table (  
TaxpayerID INT NOT NULL,  
StockID VARCHAR (255) NOT NULL,  
Quantity VARCHAR(255) NOT NULL,  
CONSTRAINT TaxpayerID_fk FOREIGN KEY (TaxpayerID) REFERENCES client_table (TaxPayerID) on delete cascade,  
CONSTRAINT StockID_fk FOREIGN KEY (StockID) REFERENCES stock_table (StockId) on delete cascade );
```

```
--INSERTING VALUES INTO CLIENTSTOCK TABLE
```

```
INSERT INTO clientstock_table (  
TaxpayerID,  
StockID,  
Quantity)  
VALUES  
((SELECT TaxpayerID  
FROM client_table WHERE TaxpayerID = 545786625),  
(SELECT StockID  
FROM stock_table WHERE StockID = 12345),  
'400');
```

```
INSERT INTO clientstock_table (  
TaxpayerID,  
StockID,  
Quantity)  
VALUES  
((SELECT TaxpayerID  
FROM client_table WHERE TaxpayerID = 123456789),  
(SELECT StockID  
FROM stock_table WHERE StockID = 67890),  
'200');
```

```
INSERT INTO clientstock_table (  
TaxpayerID,  
StockID,  
Quantity)  
VALUES  
((SELECT TaxpayerID  
FROM client_table WHERE TaxpayerID = 987654321),  
(SELECT StockID  
FROM stock_table WHERE StockID = 11234),  
'350');
```

```
INSERT INTO clientstock_table (  
TaxpayerID,  
StockID,  
Quantity)  
VALUES  
((SELECT TaxpayerID  
FROM client_table WHERE TaxpayerID = 128765930),  
(SELECT StockID  
FROM stock_table WHERE StockID = 56789),  
'660');
```

```
INSERT INTO clientstock_table (  
TaxpayerID,  
StockID,  
Quantity)  
VALUES  
((SELECT TaxpayerID  
FROM client_table WHERE TaxpayerID = 943270000),  
(SELECT StockID  
FROM stock_table WHERE StockID = 21345),  
'450');
```

--CREATING MUTUAL TABLE

```
CREATE TABLE mutual_table (  
  MutualFundID VARCHAR(255) NOT NULL,  
  FundName VARCHAR (255) NOT NULL,  
  PrincipalObjective VARCHAR (1) NOT NULL,  
  PriceRangeHigh Decimal(38,2) NOT NULL,  
  PrinceRangeLow Decimal(38,2) NOT NULL,  
  CurrentOffering Decimal(38,2) NOT NULL,  
  Yield Decimal(38,2) NOT NULL,  
  FamilyFundID INT,  
  CONSTRAINT MutualFundID_pk PRIMARY KEY (MutualFundID),  
  CONSTRAINT PrincipalObjective_Check CHECK (PrincipalObjective='G' OR PrincipalObjective='I' OR PrincipalObjective='G') );
```

--INSERTING VALUES INTO MUTUAL TABLE

```
INSERT INTO mutual_table (  
  MutualFundID,  
  FundName,  
  PrincipalObjective,  
  PriceRangeHigh,  
  PrinceRangeLow,  
  CurrentOffering,  
  Yield,  
  FamilyFundID )  
VALUES  
(  
'2345',  
'ABC Fund',  
'G',  
5500.67,  
1100.89,  
4700.11,  
12,  
110  
);
```

```
INSERT INTO mutual_table (  
  MutualFundID,
```



```
FundName,  
PrincipalObjective,  
PriceRangeHigh,  
PrinceRangeLow,  
CurrentOffering,  
Yield,  
FamilyFundID )  
VALUES  
( '7675',  
  'DEF Fund',  
  'G',  
  6500.77,  
  1700.19,  
  3400.19,  
  14,  
  111  
);
```

```
INSERT INTO mutual_table (  
  MutualFundID,  
  FundName,  
  PrincipalObjective,  
  PriceRangeHigh,  
  PrinceRangeLow,  
  CurrentOffering,  
  Yield,  
  FamilyFundID )  
VALUES  
( '8765',  
  'GHI Fund',  
  'T',  
  2200,  
  1900,  
  2050,  
  58,  
  112
```

```
);
```

```
INSERT INTO mutual_table (  
  MutualFundID,  
  FundName,  
  PrincipalObjective,  
  PriceRangeHigh,  
  PrinceRangeLow,  
  CurrentOffering,  
  Yield,  
  FamilyFundID )  
VALUES  
(5456,  
'JKL Fund',  
'T',  
1200,  
1050,  
1050,  
87,  
113  
);
```

```
INSERT INTO mutual_table (  
  MutualFundID,  
  FundName,  
  PrincipalObjective,  
  PriceRangeHigh,  
  PrinceRangeLow,  
  CurrentOffering,  
  Yield,  
  FamilyFundID )  
VALUES  
(4956,  
'XYZ Fund',  
'G',  
8200,
```

```
7600,  
7900,  
80,  
114  
);
```

```
--CREATING FAMILY TABLE  
CREATE TABLE family_table (  
FamilyFundID INT NOT NULL,  
MutualFundID VARCHAR (255) NOT NULL,  
FullName VARCHAR (255) NOT NULL,  
Street VARCHAR (255) NOT NULL,  
City VARCHAR (255) NOT NULL,  
State VARCHAR (255) NOT NULL,  
Zip INT NOT NULL,  
Country VARCHAR (255) NOT NULL,  
CONSTRAINT FamilyFundID_pk PRIMARY KEY (FamilyFundID) ,  
CONSTRAINT MutualFundID_fk FOREIGN KEY (MutualFundID) REFERENCES mutual_table (MutualFundID) on delete  
cascade);
```

```
--INSERTING VALUES IN FAMILY TABLE  
INSERT INTO family_table (  
FamilyFundID,  
MutualFundID,  
FullName,  
Street,  
City,  
State,  
Zip,  
Country  
)  
VALUES  
((SELECT FamilyFundID  
FROM mutual_table WHERE FamilyFundID = 110),
```

```
(SELECT MutualFundID
FROM mutual_table WHERE MutualFundID = 2345),
'Jose Ulloa',
'435 Fake Street',
'Los Angeles',
'CA',
'12345',
'USA');
```

```
INSERT INTO family_table (
FamilyFundID,
MutualFundID,
FullName,
Street,
City,
State,
Zip,
Country
)
VALUES
((SELECT FamilyFundID
FROM mutual_table WHERE FamilyFundID = 111),
(SELECT MutualFundID
FROM mutual_table WHERE MutualFundID = 7675),
'Mike Smith',
'1878 Noreal Blvd',
'Los Angeles',
'CA',
'45678',
'USA');
```

```
INSERT INTO family_table (
FamilyFundID,
MutualFundID,
FullName,
Street,
```

```
City,  
State,  
Zip,  
Country  
)  
VALUES  
((SELECT FamilyFundID  
FROM mutual_table WHERE FamilyFundID = 112),  
(SELECT MutualFundID  
FROM mutual_table WHERE MutualFundID = 8765),  
'Eric Zhu',  
'9898 Happy Dr',  
'Miami',  
'FL',  
'76543',  
'USA');
```

```
INSERT INTO family_table (  
FamilyFundID,  
MutualFundID,  
FullName,  
Street,  
City,  
State,  
Zip,  
Country  
)  
VALUES  
((SELECT FamilyFundID  
FROM mutual_table WHERE FamilyFundID = 113),  
(SELECT MutualFundID  
FROM mutual_table WHERE MutualFundID = 5456),  
'Aron Mejia',  
'95 Spark Street',  
'Seattle',  
'WA',
```

```
'88765',  
'USA');
```

```
INSERT INTO family_table (  
FamilyFundID,  
MutualFundID,  
FullName,  
Street,  
City,  
State,  
Zip,  
Country  
)  
VALUES  
((SELECT FamilyFundID  
FROM mutual_table WHERE FamilyFundID = 114),  
(SELECT MutualFundID  
FROM mutual_table WHERE MutualFundID = 4956),  
'Wendy Xiong',  
'765 High Street',  
'Austin',  
'TX',  
'12342',  
'USA');
```

```
--ALTERING MUTUAL TABLE  
ALTER TABLE mutual_table  
ADD CONSTRAINT FamilyFundID_fk  
FOREIGN KEY (FamilyFundID)  
REFERENCES family_table (FamilyFundID)  
on delete cascade ;
```

```
--CREATING CLIENTMUTUAL TABLE  
CREATE TABLE clientmutual_table (  
TaxpayerID INT NOT NULL,  
MutualFundID VARCHAR (255) NOT NULL,
```

```
Quantity VARCHAR(255) NOT NULL,  
CONSTRAINT TaxpayerID_fk1 FOREIGN KEY (TaxpayerID) REFERENCES client_table (TaxPayerID) on delete cascade ,  
CONSTRAINT MutualFundID_fk1 FOREIGN KEY (MutualFundID) REFERENCES mutual_table (MutualFundId) on delete  
cascade );
```

```
--INSERTING VALUES IN CLIENTMUTUAL TABLE  
INSERT INTO clientmutual_table  
(TaxpayerID,  
MutualFundID,  
Quantity)  
VALUES  
((SELECT TaxPayerID  
FROM client_table WHERE TaxPayerID = 545786625),  
(SELECT MutualFundID  
FROM mutual_table WHERE MutualFundID = 2345),  
100);
```

```
INSERT INTO clientmutual_table  
(TaxpayerID,  
MutualFundID,  
Quantity)  
VALUES  
((SELECT TaxPayerID  
FROM client_table WHERE TaxPayerID = 123456789),  
(SELECT MutualFundID  
FROM mutual_table WHERE MutualFundID = 7675),  
150);
```

```
INSERT INTO clientmutual_table  
(TaxpayerID,  
MutualFundID,  
Quantity)  
VALUES  
((SELECT TaxPayerID
```

```
FROM client_table WHERE TaxPayerID = 987654321),
(SELECT MutualFundID
FROM mutual_table WHERE MutualFundID = 8765),
200);
```

```
INSERT INTO clientmutual_table
(TaxpayerID,
MutualFundID,
Quantity)
VALUES
((SELECT TaxPayerID
FROM client_table WHERE TaxPayerID = 128765930),
(SELECT MutualFundID
FROM mutual_table WHERE MutualFundID = 5456),
250);
```

```
INSERT INTO clientmutual_table
(TaxpayerID,
MutualFundID,
Quantity)
VALUES
((SELECT TaxPayerID
FROM client_table WHERE TaxPayerID = 943270000),
(SELECT MutualFundID
FROM mutual_table WHERE MutualFundID = 4956),
300);
```

/* DESCRIBE TABLES */

```
DESC client_table;
```

```
Name      Null?  Type
-----
```

```
TAXPAYERID NOT NULL NUMBER(38)
FIRSTNAME  NOT NULL VARCHAR2(255)
LASTNAME   NOT NULL VARCHAR2(255)
```


STREET NOT NULL VARCHAR2(255)
CITY NOT NULL VARCHAR2(255)
STATE NOT NULL VARCHAR2(255)
ZIP NOT NULL NUMBER(38)
COUNTRY NOT NULL VARCHAR2(255)

DESC stock_table;

Name	Null?	Type
STOCKID	NOT NULL	VARCHAR2(255)
ISSUENAME	NOT NULL	VARCHAR2(255)
RATING	NOT NULL	VARCHAR2(100)
PRINCIPALBUINESS	NOT NULL	VARCHAR2(255)
PRICERANGEHIGH	NOT NULL	NUMBER(38,2)
PRINCERANGELOW	NOT NULL	NUMBER(38,2)
CURRENTPRICE	NOT NULL	NUMBER(38,2)
PRIORYEARRETURN		NUMBER(38,2)
FIVEYEARPERIODRETURN		NUMBER(38,2)

DESC clientstock_table;

Name	Null?	Type
TAXPAYERID	NOT NULL	NUMBER(38)
STOCKID	NOT NULL	VARCHAR2(255)
QUANTITY	NOT NULL	VARCHAR2(255)

DESC mutual_table;

Name	Null?	Type
MUTUALFUNDID	NOT NULL	VARCHAR2(255)
FUNDNAME	NOT NULL	VARCHAR2(255)

PRINCIPALOBJECTIVE NOT NULL VARCHAR2(1)
PRICERANGEHIGH NOT NULL NUMBER(38,2)
PRINCERANGELOW NOT NULL NUMBER(38,2)
CURRENTOFFERING NOT NULL NUMBER(38,2)
YIELD NOT NULL NUMBER(38,2)
FAMILYFUNDID NUMBER(38)

DESC family_table;

Name Null? Type

FAMILYFUNDID NOT NULL NUMBER(38)
MUTUALFUNDID NOT NULL VARCHAR2(255)
FULLNAME NOT NULL VARCHAR2(255)
STREET NOT NULL VARCHAR2(255)
CITY NOT NULL VARCHAR2(255)
STATE NOT NULL VARCHAR2(255)
ZIP NOT NULL NUMBER(38)
COUNTRY NOT NULL VARCHAR2(255)

DESC clientmutual_table;

Name Null? Type

TAXPAYERID NOT NULL NUMBER(38)
MUTUALFUNDID NOT NULL VARCHAR2(255)
QUANTITY NOT NULL VARCHAR2(255)

/* SELECT TABLE STATEMENTS */

SELECT * FROM client_table;

TAXPAYERID	FIRSTNAME	LASTNAME	STREET	CITY	STATE	ZIP	COUNTRY
545786625	Henry	King	1234 Fake Street	Los Angeles	CA	91754	USA
123456789	Geroge	Hill	1324 No Street	Los Angeles	CA	91753	USA
987654321	King	Pin	1673 Narrow Street	Miami	FL	12345	USA
128765930	Bob	Marley	666 High Blvd	Compton	CA	66660	USA
943270000	Nick	Sanders	123 Green Blvd	Seattle	WA	88765	USA
987765401	Donald	Bautista	556 Tall Ave	Tampa	FL	98987	USA

SELECT * FROM stock_table;

STOCKID	ISSUENAME	RATING	PRINCIPALBUINESS	PRICERANGEHIGH	PRINCERANGELOW	CURRENTPRICE	PRIORYEARRETURN	FIVEYEARPERIODRETURN
12345	ABC	7	Retail	123	100	102	109	108
67890	MNO	4	Wholeseller	787	710	778	750	767
11234	PQR	6	Restraunt	500	400	450	455	480
56789	STU	2	Stockist	900	700	800	702	710
21345	XYZ	1	Cold Storage	1100	1050	1090	1080	1093

SELECT * FROM clientstock_table;

TAXPAYERID	STOCKID	QUANTITY
545786625	12345	400
123456789	67890	200
987654321	11234	350
128765930	56789	660
943270000	21345	450

SELECT * FROM mutual_table;

MUTUALFUNDID	FUNDNAME	PRINCIPALOBJECTIVE	PRICERANGEHIGH	PRINCERANGELOW	CURRENTOFFERING	YIELD	FAMILYFUNDID
2345	ABC Fund	G	5500.67	1100.89	4700.11	12	110
7675	DEF Fund	G	6500.77	1700.19	3400.19	14	111
8765	GHI Fund	I	2200	1900	2050	58	112
5456	JKL Fund	I	1200	1050	1050	87	113
4956	XYZ Fund	G	8200	7600	7900	80	114

SELECT * FROM family_table;

FAMILYFUNDID	MUTUALFUNDID	FULLNAME	STREET	CITY	STATE	ZIP	COUNTRY
110	2345	Jose Ulloa	435 Fake Street	Los Angeles	CA	12345	USA
111	7675	Mike Smith	1878 Noreal Blvd	Los Angeles	CA	45678	USA
112	8765	Eric Zhu	9898 Happy Dr	Miami	FL	76543	USA
113	5456	Aron Mejia	95 Spark Street	Seattle	WA	88765	USA
114	4956	Wendy Xiong	765 High Street	Austin	TX	12342	USA

SELECT * FROM clientmutual_table;

TAXPAYERID	MUTUALFUNDID	QUANTITY
545786625	2345	100
123456789	7675	150
987654321	8765	200
128765930	5456	250
943270000	4956	300

/* DELETE STATEMENTS */

delete from client_table where Taxpayerid=987765401;

TAXPAYERID	FIRSTNAME	LASTNAME	STREET	CITY	STATE	ZIP	COUNTRY
545786625	Henry	King	1234 Fake Street	Los Angeles	CA	91754	USA
123456789	Geroge	Hill	1324 No Street	Los Angeles	CA	91753	USA
987654321	King	Pin	1673 Narrow Street	Miami	FL	12345	USA
128765930	Bob	Marley	666 High Blvd	Compton	CA	66660	USA
943270000	Nick	Sanders	123 Green Blvd	Seattle	WA	88765	USA

delete from stock_table where Rating=7;

STOCKID	ISSUENAME	RATING	PRINCIPALBUSINESS	PRICERANGEHIGH	PRICERANGELOW	CURRENTPRICE	PRIORYEARRETURN	FIVEYEARPERIODRETURN
67890	MNO	4	Wholeseller	787	710	778	750	767
11234	PQR	6	Restraunt	500	400	450	455	480
56789	STU	2	Stockist	900	700	800	702	710
21345	XYZ	1	Cold Storage	1200	1050	1090	1080	1093

delete from mutual_table where Mutualfundid=4956;

MUTUALFUNDID	FUNDNAME	PRINCIPALOBJECTIVE	PRICERANGEHIGH	PRICERANGELOW	CURRENTOFFERING	YIELD	FAMILYFUNDID
2345	ABC Fund	G	5500.67	1100.89	4700.11	12	110
7675	DEF Fund	G	6500.77	1700.19	3400.19	14	111
8765	GHI Fund	I	2200	1900	2050	58	112
5456	JKL Fund	I	1200	1050	1050	87	113

/* UPDATE STATEMENTS */

-- Update stock_table set pricerangehigh=1300 where stockid=21345;

Before:

STOCKID	ISSUENAME	RATING	PRINCIPALBUSINESS	PRICERANGEHIGH	PRINCERANGELOW	CURRENTPRICE	PRIORYEARRETURN	FIVEYEARPERIODRETURN
12345	ABC	7	Retail	123	100	102	109	108
67890	MNO	4	Wholeseller	787	710	778	750	767
11234	PQR	6	Restraunt	500	400	450	455	480
56789	STU	2	Stockist	900	700	800	702	710
21345	XYZ	1	Cold Storage	1100	1050	1090	1080	1093

After: select * from stock_table where stockid=21345;

STOCKID	ISSUENAME	RATING	PRINCIPALBUSINESS	PRICERANGEHIGH	PRINCERANGELOW	CURRENTPRICE	PRIORYEARRETURN	FIVEYEARPERIODRETURN
21345	XYZ	1	Cold Storage	1300	1050	1090	1080	1093

-- Update clientstock_table set quantity=700 where stockid=11234;

Before:

TAXPAYERID	STOCKID	QUANTITY
545786625	12345	400
123456789	67890	200
987654321	11234	350
128765930	56789	660
943270000	21345	450

After: select * from clientstock_table where stockid=11234;

TAXPAYERID	STOCKID	QUANTITY
987654321	11234	700

/* VIEWS */

create or replace view client_view

AS (select taxpayerid,firstname,lastname,state,zip from client_table where taxpayerid in(943270000,987654321,123456789));

View CLIENT_VIEW created.

select * from client_view;

TAXPAYERID	FIRSTNAME	LASTNAME	STATE	ZIP
123456789	Geroge	Hill	CA	91753
943270000	Nick	Sanders	WA	88765
987654321	King	Pin	FL	12345

create or replace view family_view

As (select familyfundid, mutualfundid, fullname from Family_Table where city='Los Angeles');

View FAMILY_VIEW created.

select * from family_view;

FAMILYFUNDID	MUTUALFUNDID	FULLNAME
110	2345	Jose Ulloa
111	7675	Mike Smith

/* COMPLEX SELECT STATEMENTS, JOIN, GROUP BY, ETC DML */

```

select c.taxpayerid , c.firstname, c.lastname, c.state, s.stockid, s.quantity
from client_table c, clientstock_table s
where c.taxpayerid=s.taxpayerid
having s.quantity between 200 and 450 group by c.taxpayerid,c.firstname,c.lastname,c.state, s.stockid, s.quantity;

```

TAXPAYERID	FIRSTNAME	LASTNAME	STATE	STOCKID	QUANTITY
123456789	Geroge	Hill	CA	67890	200
943270000	Nick	Sanders	WA	21345	450

```

select * from
(select b.stockid,a.firstname,a.lastname, b.principalbuiness, b.rating, b.currentprice
from clientstock_table c,stock_table b,client_table a
where b.stockid =c.stockid and a.taxpayerid=c.taxpayerid)
where principalbuiness='Wholeseller' --- Inline views
ORDER BY currentprice

```

STOCKID	FIRSTNAME	LASTNAME	PRINCIPALBUINESS	RATING	CURRENTPRICE
67890	Geroge	Hill	Wholeseller	4	778

/*PL SQL*/

```
--Raise stock quantity by 10% for TaxpayerID
CREATE OR REPLACE PROCEDURE raise_stock
    (v_id in clientstock_table.TaxpayerID%type)
IS
BEGIN
    UPDATE clientstock_table
    SET Quantity= Quantity *1.10
    WHERE TaxpayerID =v_id;
END raise_stock;
```

```
EXECUTE raise_stock (123456789)
```

```
-- CREATE OR REPLACE PROCEDURE raise_currentprice
(v_id in stock_table.CurrentPrice%type)
IS
BEGIN
    UPDATE stock_table
    SET CurrentPrice = CurrentPrice *.80
    WHERE CurrentPrice =v_id;
END raise_currentprice;
```

```
EXECUTE raise_currentprice (102);
```

```
-- Raise HighPrice by 300 as per rating
```

```
CREATE OR REPLACE PROCEDURE raise_highprice
    (v_rating in stock_table.rating%type)
IS
BEGIN
    update stock_table
    set Pricerangehigh=Pricerangehigh + 300
    where rating=v_rating;
END raise_highprice;
```

```
EXECUTE raise_highprice(2);
```

```
--CREATE OR REPLACE PROCEDURE query_clientstock
```

```
CREATE OR REPLACE PROCEDURE query_clientstock
```

```
(v_id IN clientstock_table.taxpayerid%TYPE,  
  v_id2 OUT clientstock_table.taxpayerid%TYPE,  
  v_quantity OUT clientstock_table.quantity%TYPE)
```

```
IS
```

```
BEGIN
```

```
  SELECT taxpayerid,quantity  
  INTO v_id2, v_quantity  
  from clientstock_table  
  WHERE taxpayerid =v_id;  
END query_clientstock;
```

References

SQL Tutorial. (n.d.). Retrieved April 15, 2018, from <https://www.w3schools.com/sql/>

What is SQL (Structured Query Language)?. (n.d.) Retrieved April 15, 2018, from <https://searchsqlserver.techtarget.com/definition/SQL>