



# **An Analysis on Stock Market using Database Management System**

*to be submitted in partial fulfilling of the requirements for the course  
on*

**Database Management Systems – ITE 1003**

**(A1 & A2)**

**Group members:**

Virat Singh Chauhan [18BIT0090]

Aryan Khandelwal [18BIT171]

**Under the Guidance of:** Prof. Dharmendra Singh Rajput



Fall Semester 2020-2021

## TABLE OF CONTENTS

<b>1. Introduction &amp; Abstract</b>	.....
<b>2. Review 1 (Survey, Analysis).....</b>	
<b>3. Review 2 (Design of Diagrams &amp; Prototype Design).....</b>	
<b>4. Review 3 (Development of Model).....</b>	
<b>5. Conclusion .....</b>	
<b>6. References .....</b>	

## Abstract

The objective of this project is to develop a Stock Analysis system that is basically for the Investors coming into the domain of Stock Market where they can learn about how the stock market works based on the previous data which is already fed into the system.

The exchange is seen as associate degree everyday a part of life in any economy is it developed or assemblage countries. because the world is step by step moving to the data age that a significant characteristic of this age is speed; speed in causation messages, speed in health care delivery, speed in banking, there's a necessity for investors to be able to monitor their investments with speed. during this work, we tend to introduce P-Stock, associate degree application for observance the activities that maintain within the exchange. It tracks exchange activities likewise as mistreatment the info that has been pooled to calculate the values of stock in associate degree investor's portfolio bushed time period. it's hoped that this may facilitate new investors in observance their investments primary while not would like for his or her brokers.

**KEYWORDS:** Database, Stock Analysis, Sensex, User, stock market; stock observance

## Introduction

In recent years the study of data analysis has apparently moved from collection and storage topics to more descriptive and scientific issues. Thus, to study the data analysis approaches, we have found the patterns of research and transferred to the popular subjects. EMC Digital World Research has pointed out that global data capacity is a field which is in a state of constant change. The quantities of data are increased from 4.4ZB to 44ZB, the effective information from the retrieval data becomes a significant concern in a wide variety such as the investments of stock, fund etc.

The stock exchange represents the key institution for the development of capital market of any country. Thus the information system of every stock exchange must satisfy very strict international standards.

Stock market includes daily activities like sensdex calculation, exchange of shares. The exchange provides an efficient and transparent market for trading in equity, debt instruments and derivatives. Analysis of stocks using database management systems will be useful for new investors to invest in stock market based on the various factors considered by the software.

In this paper we tend to show the quality and feasibleness of applying typical SQL queries for analysing a good spectrum of information streams. As application space we've chosen the analysis of stock exchange knowledge, principally as a result of this sort of application exhibits sufficiently several of these characteristics that relative question technology may be thought of a valuable instrument during a stream context. The ensuing TInTo system may be a tool for computing questionable technical indicators, numerical values calculated from an explicit reasonably stock exchange knowledge, characterizing the event of stock costs over a given period. Update propagation is employed for the progressive re-computation of indicator views outlined over a stream of incessantly dynamical worth knowledge.

## Data Requirements

- **Index**

Role:- The Companies are listed under some type of index generally in India in NSE(National Stock Exchange) or BSE(Bombay Stock Exchange)

- └ Registration number as Primary Key

- └ Date of Listing

- **Company**

Role: Corporations in which the liability of each shareholder is limited to the amount individually invested. The most common form of the company used for business ventures.

- └ C\_NAME

- └ REVENUE

- **Role:**

In accounting, revenue is the income that a business has from its normal business activities, usually from the sale of goods and services to customers.

Revenue is also referred to as sales or turnover. Some companies receive revenue from interest, royalties, or other fees.

- └ PROFIT

- └ EXPENSES

- **SHARES**

Role: In financial markets, a share is a unit used as mutual funds, limited partnerships, and real estate investment trusts. The owner of shares in the company is a shareholder of the corporation. A share is an indivisible unit of capital, expressing the ownership relationship between the company and the shareholder.

- └ S\_NAME

- └ NO. OF SHARES

- └ LOT SIZE

Role: In terms of stocks, the lot is the number of shares you purchase in one transaction.

- └ FACE VALUE

Role: The face value, also known as par value, is the legal capital of the share.

- **CUSTOMER**

Role: A customer is an individual or institution that legally owns one or more shares of stock in a public or private corporation. Shareholders may be referred to as members of a corporation.

- └ NAME

- └ NO. OF SHARES PURCHASED

- └ C\_ID

- └ PH\_NO

- └ **BROKER**

Role: The main role of the broker is to manage other customer while buying the shares

- └ BROKER\_ID

- └ **CURRENCY**

Role: The Currency is the basis for the Transaction

- └ CODE

- └ NAME

- └ IS\_BASE

- └ IS\_ACTIVE

- └ **BILL**

- └ BILL ID

- └ BILL DATE

- └ BILL\_DESCN

**18BIT0171**

**18BIT0090**

- 1) All the customers must be provided with the C\_Id and password for them.
- 2) A customer can purchase any number of shares in a company.
- 3) A share can be purchased jointly by one or more customers.
- 4) Transactions is a weak entity type with foreign key C\_Id
- 5) A Company gets registered in the primary market to provide a share to the public and raise funds.
- 6) The shares are identified by S\_name while companies are identified by C\_name.
- 7) Admin will be provided with a username and password using them he can manage the database.
- 8) Index is Identified by Companies Registration number
- 9) Index is Related as Indexing the company in the Share Market

The Market lot is the amount of shares to be provided by the company for the minimum transaction amount of shares allowed for the share transactions.

**18BIT0171**  
**18BIT0090**

## **Functional Requirements**

### **Company**

Requirement	Description
C01	The company shall be able to register in the primary market to provide a share to the public and raise funds.
C02	The company shall be able to see the statistical data view of stock market analysis to predict the investor's opinion of itself.

### **Shares**

Requirement	Description
S01	The Current Fluctuation in the Share price can be controlled by the Regulatory
S02	The amount of shares issued by the company shall be updated by the company while listing it on the Index

### **Customer**

Requirement	Description
U01	The customer shall be provided with C_Id and password to login to access the database.
U02	The customer shall be able to inquire about a particular company stock market data.
U03	The customer shall be able to view the statistical depiction of the stock analysis.



**18BIT0171**  
**18BIT0090**

### **Index**

Requirement	Description
I01	Index shall decide where the stock has to be listed and in which region its transactions will be shown.

### **Broker**

Requirement	Description
I01	Index shall decide where the stock has to be listed and in which region its transactions will be shown.

### **Scenarios of Removal Of Old data**

└

**When the customer sells the shares it has to be removed from the customer base.**

└

**When the company is unlisted it has to be removed from the database**

### **Scenarios of Data Modification**

└

**When the market value of Share is Decreased or Increased accordingly we have to modify its current price**

└

**When the company issues new shares the current share value has to be modified**

### **Scenarios of Data Retrieval**

└

**No. Of Shares currently held by the customer**

└

**Net Profit/Loss for the Customer**

└

**Date of Purchase or Sell of Shares for the Customer**

└

**Company details its Current market value and No. of Shares**

### NON FUNCTIONAL REQUIREMENTS:

1. **Reliability:** The reliability of the product will be dependent on the accuracy of the data, date of purchase, how much stock was purchased, high and low value range as well as opening and closing figures.
2. **Security:** The user will only be able to access the software using his login details and will not be able to access the computations happening at the back end.
3. **Maintainability:** The maintenance of the product would require the database to be updated with recent values.
4. **Portability:** The software is completely portable.
5. **Interoperability:** The interoperability of the software is very high because it synchronizes all the database with the server.

### OUTPUT DESIGN:

Outputs from stock market analysis systems are required primarily to communicate the results of analysis to users. They are also used to provide a permanent copy of the results for later consultation.

The outputs should be defined in terms of the following points:

- Type of the output
  - Content of the output
  - Format of the output
  - Frequency of the output
  - Volume of the output
  - Sequence of the output

### INPUT DESIGN:

#### INPUT TYPES:

It is necessary to determine the various types of inputs.

Inputs can be categorized as follows:

- External inputs, which are prime inputs for the stock market analysis system..
- Internal inputs, which are user communications with the system.
- Interactive, which are inputs entered during a dialogue.

### **INPUT MEDIA:**

At this stage choice has to be made about the input media. To conclude about the input media consideration has to be given to:

- ┐ Type of input
- ┐ Flexibility of format
- ┐ Speed
- ┐ Accuracy
- ┐ Verification methods
- ┐ Rejection rates
- ┐ Ease of correction
- ┐ Storage and handling requirements
- ┐ Security
- ┐ Easy to use
- Portability

Keeping in view the above description of the input types and input media, it can be said that most of the inputs are of the form of external and interactive. As input data is to be the directly keyed in by the user, the keyboard can be considered to be the most suitable input device. For external input i.e the raw dataset of old stock market values of a company is collected from various websites.

### **ERROR AVOIDANCE:**

At this stage care is to be taken to ensure that input data remains accurate from the stage at which it is recorded upto the stage in which the data is accepted by the system. This can be achieved only by means of careful control each time the data is handled.

### **ERROR DETECTION:**

Even though every effort is made to avoid the occurrence of errors, still a small proportion of error is always likely to occur, these types of errors can be discovered by using validations to check the input data.

### **DATA VALIDATION OF CUSTOMER:**

Procedures are designed to detect errors in data at a lower level of detail. Data validations have been included in the system in almost every area where there is a possibility for the user to commit errors.

## **USER INTERFACE DESIGN**

### **COMPUTER-INITIATED INTERFACES:**

The following computer – initiated interfaces are used:

1. The Stock market query management system for the user is presented with an analysis of the company's shares which enables investors to identify the intrinsic worth of a security even before investing in it.
2. Statistical Data Analysis – Pie Charts and Bar Graphs are used to statistically depict the analysis of a particular company's stock market of which the user has inquired about.

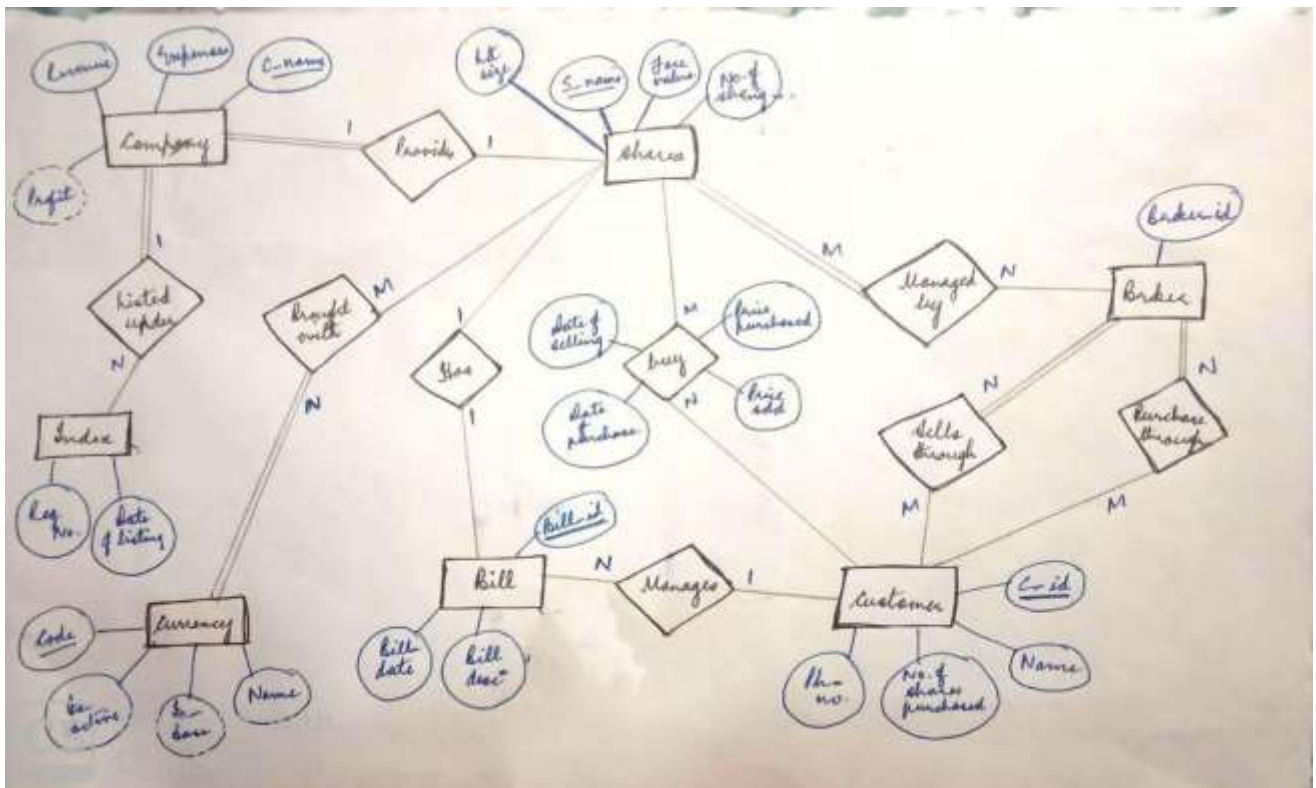
### **ERROR MESSAGE DESIGN:**

The design of error messages is an important part of the user interface design. As user is bound to commit some errors or other while designing a stock market query management system it should be implemented in such a way that it is helpful by providing the user with information regarding the error he/she has committed.

18BIT0171

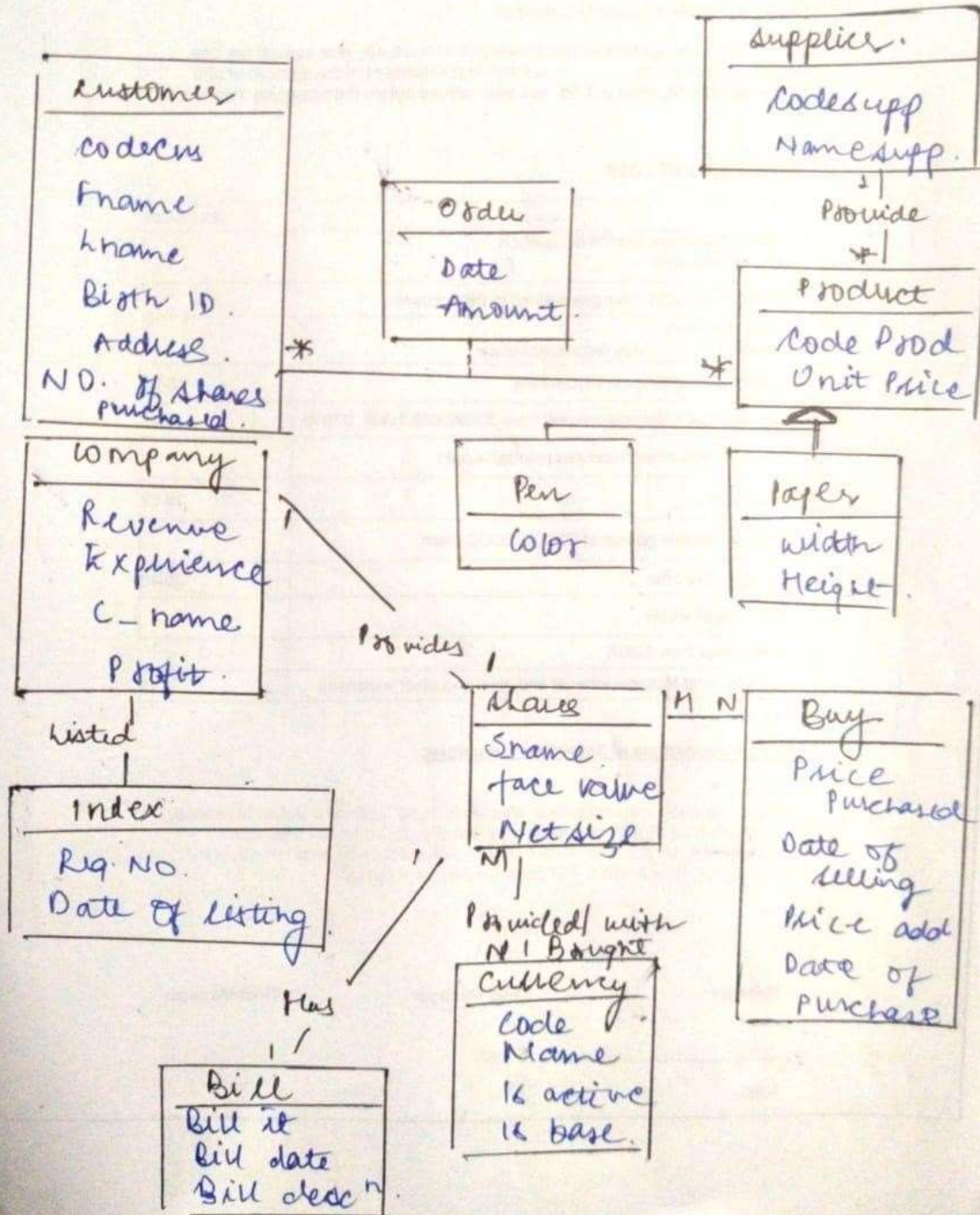
18BIT0090

E.R DIAGRAM:



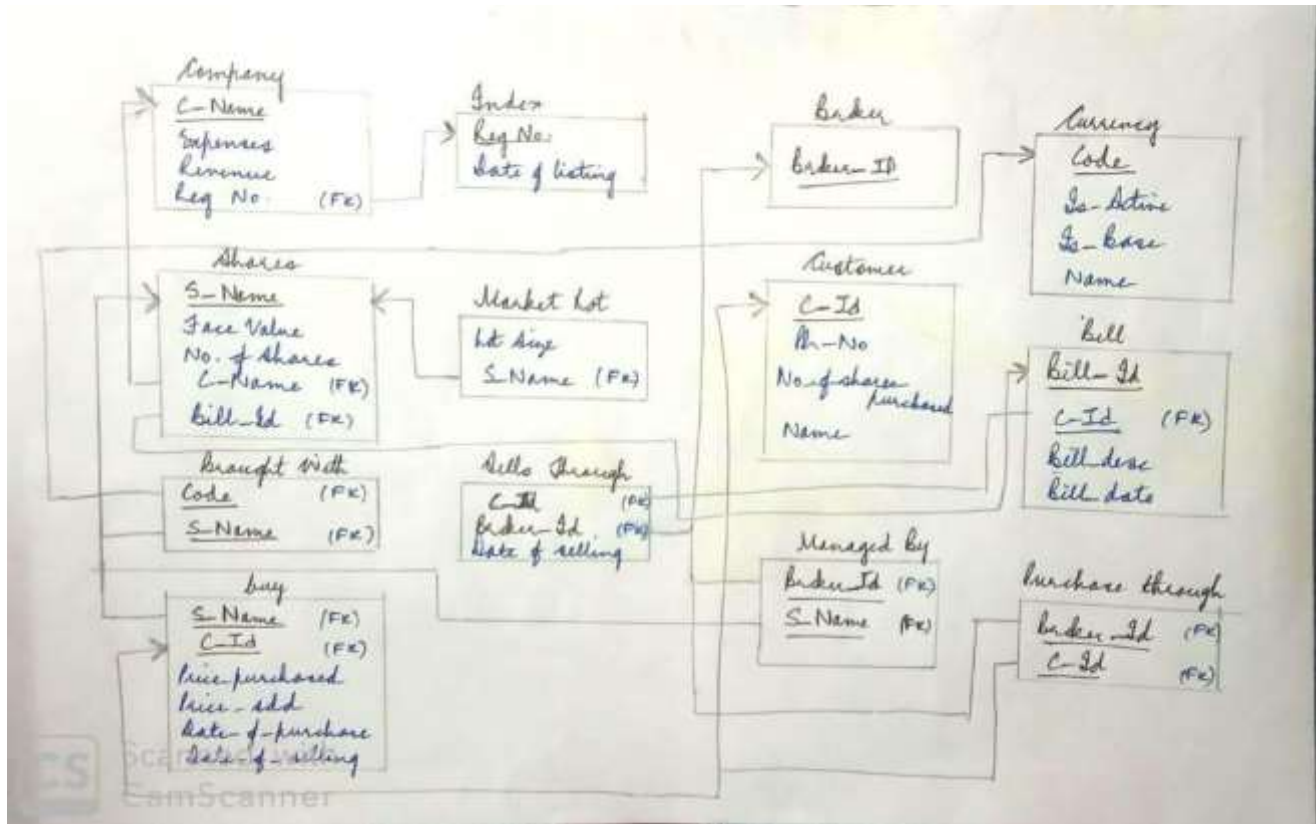
# Relational Schema:

The VML conceptual schema of  
Stock Management System.





# RELATIONSHIP SCHEMA



**Database:**

```
SQL> create table index_proj(  
2 reg_no Number(5) primary key,  
3 date_of_listing date);
```

Table created.

```
SQL> create table company_proj(  
2 c_name varchar(20) primary key,  
3 expenses number(9),  
4 revenue number(9),  
5 reg_no references index_proj);
```

Table created.

```
SQL> create table broker_proj(  
2 broker_id number(5) primary key  
3 );
```

Table created.

```
SQL> create table shares(  
2
```

```
SQL> create table bill_proj(  
2 bill_id number(5) primary key,  
3 bill_cust_id number(5),  
4 bill_descn varchar(20),  
5 bill_date date);
```

Table created.

```
SQL> create table currency_proj(  
2 code varchar(5) primary key, 3  
is_base varchar(5),  
4 is_active varchar(5),  
5 name varchar(20)  
6 );
```

Table created.



```
SQL> create table shares_proj(  
2 s_name varchar(20) primary key,  
3 face_value number(6),  
4 no_of_shares number(10),  
5 c_name references company_proj,  
6 bill_id references bill_proj);
```

Table created.

```
SQL> create table brought_with_proj(  
2 code references currency_proj,  
3 s_name references shares_proj  
4 );
```

Table created.

```
SQL> create table customer_proj(  
2 c_id number(7) primary key,  
3 ph_no number(10),  
4 no_of_shares_purchased number(10),  
5 name varchar(20),  
6 Bill_id references
```

bill\_proj); Table created.

```
SQL> create table managed_by_proj(  
2 broker_id references broker,  
3
```

```
SQL> create table managed_by_proj(  
2 broker_id references broker_proj, 3  
s_name references shares_proj);
```

Table created.

**18BIT0171**  
**18BIT0090**

```
SQL> create table purchases_through_proj(  
  2 broker_id references broker_proj,  
  3 c_id references customer_proj);
```

Table created.

```
SQL> create table market_lot_proj(  
  2 lot_size number(4),  
  3 s_name references shares_proj);
```

Table created.

```
SQL> create table sells_through_proj(  
  2 c_id references customer_proj,  
  3 broker_id references broker_proj,  
  4 date_of_selling date);
```

Table created.

```
SQL> create table buy_proj(  
  2 s_name references shares_proj,  
  3 c_id references customer_proj,  
  4 price_purchased number(7),  
  5 price_sold number(7),  
  6 date_of_purchase date,  
  7 date_of_selling date);
```

Table created.

## 18BIT0171

## 18BIT0090

```
Command Prompt - sqlplus

Microsoft Windows [Version 10.0.17763.805]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\HIDM0088> sqlplus

SQL*Plus: Release 11.2.0.2.0 Production on Tue Oct 15 00:45:19 2019
Copyright (c) 2002, 2014, Oracle. All rights reserved.

Enter user-name: system
Enter password:

Connected to:
Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit Production

SQL> create table index_proj(
  2 reg_no number(5) primary key,
  3 date_of_listing date);

Table created.

SQL> create table company_proj(
  2 c_name varchar(20) primary key,
  3 expenses number(5),
  4 revenue number(5),
  5 reg_no references index_proj);

Table created.

SQL> create table broker(
  2 broker_id number(5) primary key
  3 );

create table broker(
  4 );

ERROR at line 1:
ORA-00055: name is already used by an existing object

SQL> create table broker_proj(
  2 broker_id number(5) primary key
  3 );

Table created.

SQL> create table shares(
  2 );

SQL> create table bill_proj(
  2 bill_id number(5) primary key,
  3 bill_cost_id number(5),
  4 bill_desc varchar(20),
  5 bill_date date);

Table created.
```

```
Command Prompt - sqlplus

SQL> create table bill_proj(
  2 bill_id number(5) primary key,
  3 bill_cost_id number(5),
  4 bill_desc varchar(20),
  5 bill_date date);

Table created.

SQL> create table currency(
  2 code number(5) primary key,
  3 is_base varchar(5),
  4 is_active varchar(5),
  5 name varchar(20)
  6 );

SQL> create table currency(
  2 code varchar(5) primary key,
  3 is_base varchar(5),
  4 is_active varchar(5),
  5 name varchar(20)
  6 );

create table currency(
  7 );

ERROR at line 1:
ORA-00055: name is already used by an existing object

SQL> create table currency_proj(
  2 code varchar(5) primary key,
  3 is_base varchar(5),
  4 is_active varchar(5),
  5 name varchar(20)
  6 );

Table created.

SQL> create table shares_proj(
  2 s_name varchar(20) primary key,
  3 face_value number(8),
  4 no_of_shares number(10),
  5 c_name references company_proj,
  6 bill_id references bill_proj);

Table created.

SQL> create table bought_with_proj(
  2 code references currency_proj,
  3 s_name references shares_proj,
  4 );

c_name references shares_proj
```

## 18BIT0171

## 18BIT0090

```
Command Prompt - sqlplus

SQL> create table bought_with_proj(
  2  c_id references currency_proj,
  3  s_name references shares_proj,
  4  );

Table created.

SQL> create table customer_proj(
  2  c_id number(7) primary key,
  3  ph_no number(10),
  4  no_of_shares_purchased number(10),
  5  name varchar(20),
  6  bill_id references bill_proj);

Table created.

SQL> create table managed_by_proj(
  2  broker_id references broker,
  3  );

SQL> create table managed_by_proj(
  2  broker_id references broker_proj,
  3  s_name references shares_proj);

Table created.

SQL> create table purchases_through_proj(
  2  broker_id references broker_proj,
  3  c_id references customer_proj);

Table created.

SQL> create table market_lot_proj(
  2  lot_size number(4),
  3  s_name references shares_proj);

Table created.

SQL> create table sells_through_proj(
  2  c_id references customer_proj,
  3  broker_id references broker_proj,
  4  date_of_selling date);

Table created.

SQL> create table buy_proj(
  2  s_name references shares_proj,
  3  c_id references customer_proj,
  4  price_purchased number(7),
  5  price_sold number(7),
  6  date_of_purchase date);

Type here to search
```

```
Command Prompt - sqlplus

SQL> create table managed_by_proj(
  2  broker_id references broker,
  3  );

SQL> create table managed_by_proj(
  2  broker_id references broker_proj,
  3  s_name references shares_proj);

Table created.

SQL> create table purchases_through_proj(
  2  broker_id references broker_proj,
  3  c_id references customer_proj);

Table created.

SQL> create table market_lot_proj(
  2  lot_size number(4),
  3  s_name references shares_proj);

Table created.

SQL> create table sells_through_proj(
  2  c_id references customer_proj,
  3  broker_id references broker_proj,
  4  date_of_selling date);

Table created.

SQL> create table buy_proj(
  2  s_name references shares_proj,
  3  c_id references customer_proj,
  4  price_purchased number(7),
  5  price_sold number(7),
  6  date_of_purchase date,
  7  date_of_selling date);

Table created.

SQL>
```

## 18BIT0171

## 18BIT0090

```
Command Prompt - sqlsh
SQL> desc company_proj
Name                                     Null?   Type
-----
C_NAME                                NOT NULL VARCHAR2(20)
CPTIMEIS                             NUMBER(9)
RSTTIME                                NUMBER(9)
RST_NO                                NUMBER(5)

SQL> desc shares_proj
Name                                     Null?   Type
-----
S_NAME                                NOT NULL VARCHAR2(20)
PRICE_VALUE                           NUMBER(6)
NO_OF_SHARES                           NUMBER(10)
C_NAME                                VARCHAR2(20)
BILL_ID                                NUMBER(5)

SQL> desc brought_with_proj
ERROR
ORA-04043: object brought_with_proj does not exist

SQL> desc brought_with_proj
Name                                     Null?   Type
-----
CODE                                  VARCHAR2(5)
S_NAME                                VARCHAR2(20)

SQL> desc buy_proj
Name                                     Null?   Type
-----
S_NAME                                VARCHAR2(20)
C_ID                                  NUMBER(7)
PRICE_PURCHASED                       NUMBER(7)
PRICE_SOLD                             NUMBER(7)
DATE_OF_PURCHASE                       DATE
DATE_OF_SELLING                       DATE

SQL> desc index_proj
Name                                     Null?   Type
-----
RST_NO                                NOT NULL NUMBER(5)
DATE_OF_LISTING                       DATE

SQL> desc market_lot_proj
Name                                     Null?   Type
-----
LOT_SIZE                              NUMBER(4)
S_NAME                                VARCHAR2(20)
```

```
Command Prompt - sqlsh
SQL> desc sells_through_proj
Name                                     Null?   Type
-----
F_ID                                  NUMBER(7)
BROKER_ID                             NUMBER(5)
DATE_OF_SELLING                       DATE

SQL> desc broker_proj
Name                                     Null?   Type
-----
BROKER_ID                             NOT NULL NUMBER(5)

SQL> desc customer_proj
Name                                     Null?   Type
-----
C_ID                                  NOT NULL NUMBER(7)
PIN_NO                                NUMBER(10)
NO_OF_SHARES_PURCHASED                NUMBER(10)
NAME                                    VARCHAR2(20)
BILL_ID                                NUMBER(5)

SQL> desc managed_by_proj
Name                                     Null?   Type
-----
BROKER_ID                             NUMBER(5)
S_NAME                                VARCHAR2(20)

SQL> desc currency_proj
Name                                     Null?   Type
-----
CODE                                  NOT NULL VARCHAR2(5)
IS_BASE                                VARCHAR2(5)
IS_ACTIVE                              VARCHAR2(5)
NAME                                    VARCHAR2(20)

SQL> desc bill_proj
Name                                     Null?   Type
-----
BILL_ID                                NOT NULL NUMBER(5)
BILL_COST_ID                          NUMBER(5)
BILL_DESC                               VARCHAR2(20)
BILL_DATE                             DATE

SQL> desc purchases_through_proj
Name                                     Null?   Type
-----
BROKER_ID                             NUMBER(5)
C_ID                                  NUMBER(7)
```

**18BIT0171**

**18BIT0090**

SQL> insert into customer\_proj values(111,8989898989,400,'ABCDEF'); 1 row  
created.

SQL> insert into bill\_proj values(1,111,'Purchase',to\_date('13-10-19','dd-mm-yy'));  
insert into bill\_proj values(1,111,'Purchase',to\_date('13-10-19','dd-mm-yy'))  
\*

ERROR at line 1:

ORA-00917: missing comma

SQL> insert into bill\_proj values(1,111,'Purchase',to\_date('13-10-19','dd-mm-yy'));  
  
1 row created.

SQL> select \* from shares\_proj; no  
  
rows selected

SQL> insert into shares\_proj values('Reliance Ltd',450,40000,'Reliance',1); 1 row  
created.

SQL> insert into market\_lot\_proj(10,'Reliance Ltd'); insert into  
market\_lot\_proj(10,'Reliance Ltd')  
\*

ERROR at line 1:

ORA-00928: missing SELECT keyword

SQL> insert into market\_lot\_proj values(10,'Reliance Ltd'); 1 row  
created.

SQL> create table brought\_with\_proj('INR','Reliance Ltd'); create  
table brought\_with\_proj('INR','Reliance Ltd')  
\*

18BIT0171  
18BIT0090

ERROR at line 1:  
ORA-00904: : invalid identifier

SQL> create table brought\_with\_proj values('INR','Reliance Ltd'); create table  
brought\_with\_proj values('INR','Reliance Ltd')

\*

ERROR at line 1:  
ORA-00922: missing or invalid option

SQL> insert into brought\_with\_proj values('INR','Reliance Ltd'); 1 row  
created.

SQL> insert into sells\_through\_proj values(111,12345,to\_date('15-10-19','dd-mm-yy'));  
  
1 row created.

SQL> insert into managed\_by\_proj values(12345,'Reliance Ltd'); 1 row  
created.

SQL> insert into purchases\_through\_proj values(12345,111); 1 row  
created.

SQL> insert into buy\_proj values('Reliance Ltd',111,525,528,to\_date('14-07-19','dd-mm-yy'),to\_date('15-08-19','dd-mm-yy'));

1 row created.

SQL> select \* from index\_proj;

```
REG_NO DATE_OF_L
-----
11111
10-OCT-10
```

SQL> select \* from company\_proj;

18BIT0171  
18BIT0090

C_NAME	EXPENSES	REVENUE	REG_NO
			Reliance
	5800000	7800000	11111

SQL> select \* from shares\_proj;

S_NAME	FACE_VALUE	NO_OF_SHARES	C_NAME	BILL_ID
-----				
Reliance Ltd	450	40000	Reliance	1

SQL> select \* from customer\_proj;

C_ID	PH_NO	NO_OF_SHARES_PURCHASED	NAME
-----			
111	8989898989	400	ABCDEF

SQL> select \* from market\_lot\_proj;

LOT_SIZE	S_NAME
-----	
10	Reliance Ltd

SQL> select \* from broker\_proj;

BROKER_ID
-----
12345

SQL> select \* from currency\_proj;

CODE	IS_BA	IS_AC	NAME
-----			
INR	Yes	Yes	Indian Rupees

SQL>

select \* from bill\_proj;

BILL_ID	BILL_CUST_ID	BILL_DESCN	BILL_DATE
-----			
1	111	Purchase	13-OCT-19



18BIT0171  
18BIT0090

SQL> select \* from brought\_with\_proj;

CODE S\_NAME

-----

INR Reliance Ltd

SQL> select \* from buy\_proj;

S\_NAME C\_ID PRICE\_PURCHASED PRICE\_SOLD  
DATE\_OF\_P DATE\_OF\_S

----- Reliance Ltd  
111 525 528 14-JUL-19 15-AUG-19

SQL> select \* from sells\_through\_proj;

C\_ID BROKER\_ID DATE\_OF\_S

-----

111 12345 15-OCT-19

SQL> select \* from managed\_by\_proj;

BROKER\_ID S\_NAME

-----

12345 Reliance Ltd

SQL> select \* from purchases\_through\_proj;

BROKER\_ID C\_ID

-----

12345 111

## 18BIT0171

## 18BIT0090

```
Command Prompt - sqlplus

ERROR at line 1:
ORA-00928: missing SELECT keyword

SQL> insert into customer_prof values(111,000000000,000,'AMERICA');
1 row created.

SQL> insert into bill_prof values(1,111,'Purchase',to_date('11-10-19','dd-mm-yy'));
insert into bill_prof values(1,111,'Purchase',to_date('11-10-19','dd-mm-yy'))
ERROR at line 1:
ORA-00917: missing comma

SQL> insert into bill_prof values(1,111,'Purchase',to_date('11-10-19','dd-mm-yy'));
1 row created.

SQL> select * from shares_prof;
no rows selected

SQL> insert into shares_prof values('Bellanca Ltd',450,00000,'Bellanca',1);
1 row created.

SQL> insert into market_bot_prof(10,'Bellanca Ltd');
insert into market_bot_prof(10,'Bellanca Ltd')
ERROR at line 1:
ORA-00928: missing SELECT keyword

SQL> insert into market_bot_prof values(10,'Bellanca Ltd');
1 row created.

SQL> create table brought_with_prof('DM','Bellanca Ltd');
create table brought_with_prof('DM','Bellanca Ltd')
ERROR at line 1:
ORA-00904: invalid identifier

SQL> create table brought_with_prof values('DM','Bellanca Ltd');
create table brought_with_prof values('DM','Bellanca Ltd')
ERROR at line 1:
ORA-00922: missing or invalid action
```

```
Command Prompt - sqlplus

BILL_ID BILL_CUST_ID BILL_DESCN BILL_DATE
-----
1 111 Purchase 11-OCT-19

SQL> select * from brought_with_prof;

CODE S_NAME
-----
DM Bellanca Ltd

SQL> select * from buy_prof;

S_NAME C_ID PRICE PURCHASED PRICE SOLD DATE OF P DATE OF S
-----
Bellanca Ltd 111 520 500 14-JUL-19 15-AUG-19

SQL> select * from sells_through_prof;

C_ID BROKER_ID DATE OF S
-----
111 12345 15-OCT-19

SQL> select * from managed_by_prof;

BROKER_ID S_NAME
-----
12345 Bellanca Ltd

SQL> select * from purchases_through_prof;

BROKER_ID C_ID
-----
12345 111

SQL>
```

**18BIT0171**  
**18BIT0090**

```
SQL> select * from  
company_proj 2 ;
```

C_NAME	EXPENSES	REVENUE	REG_NO
--------	----------	---------	--------

-----

Reliance	5800000	7800000	11111
----------	---------	---------	-------

```
SQL> insert into index_proj values(22222,to_date('10-09-2010','dd-mm-  
yyyy'); insert into index_proj values(22222,to_date('10-09-2010','dd-mm-  
yyyy')
```

\*

ERROR at line 1:

ORA-00917: missing comma

```
SQL> insert into index_proj values(22222,to_date('10-09-2010','dd-mm-yyyy'));
```

1 row created.

```
SQL> insert into index_proj values(33333,to_date('10-08-2010','dd-mm-yyyy'));
```

1 row created.

```
SQL> insert into index_proj values(44444,to_date('10-07-2010','dd-mm-yyyy'));
```

1 row created.

**18BIT0171**  
**18BIT0090**

```
SQL> insert into index_proj values(55555,to_date('10-06-2010','dd-mm-yyyy'));
```

1 row created.

```
SQL> insert into company_proj values('Shell india',4000000,7800000,22222);
```

1 row created.

```
SQL> insert into company_proj values('Shell  
india',7000000,8800000,33333); insert into company_proj values('Shell  
india',7000000,8800000,33333)
```

\*

ERROR at line 1:

ORA-00001: unique constraint (SYSTEM.SYS\_C007383) violated

```
SQL> insert into company_proj values('HPCL',7000000,8800000,33333);
```

1 row created.

```
SQL> insert into company_proj values('Balaji store',6000000,9800000,44444);
```

1 row created.

```
SQL> insert into shares_proj values('Shell ltd',300,50000,'Shell  
india'); insert into shares_proj values('Shell ltd',300,50000,'Shell  
india')
```

\*

**18BIT0171**  
**18BIT0090**

ERROR at line 1:

ORA-00947: not enough values

SQL> insert into customer\_proj values(222,8172856627,500,'Sarthak');

1 row created.

SQL> insert into customer\_proj values(333,9889073505,600,'Shivam');

1 row created.

SQL> insert into customer\_proj values(444,9140286432,700,'Harshita');

1 row created.

SQL> insert into customer\_proj values(555,9794775121,200,'Virat');

1 row created.

SQL> insert into bill\_proj values(2,222,Purchase,to\_date('13-09-2019','dd-mm-yyyy'))); insert into bill\_proj values(2,222,Purchase,to\_date('13-09-2019','dd-mm-yyyy'))

\*

ERROR at line 1:

ORA-00984: column not allowed here

**18BIT0171**  
**18BIT0090**

```
SQL> insert into bill_proj values(2,222,'Purchase',to_date('13-09-2019','dd-mm-yyyy'));
```

1 row created.

```
SQL> insert into bill_proj values(3,333,'Sell',to_date('13-08-2019','dd-mm-yyyy'));
```

1 row created.

```
SQL> insert into bill_proj values(4,444,'Purchase',to_date('13-07-2019','dd-mm-yyyy'));
```

1 row created.

```
SQL> insert into bill_proj values(5,333,'Purchase',to_date('13-06-2019','dd-mm-yyyy'));
```

1 row created.

```
SQL> insert into broker_proj values(12346);
```

1 row created.

```
SQL> insert into broker_proj values(12347);
```

1 row created.

```
SQL> insert into broker_proj values(12348);
```

1 row created.

**18BIT0171**  
**18BIT0090**

```
SQL> insert into broker_proj values(12349);
```

1 row created.

```
SQL> insert into currency values('USD','no','no','American  
Dollar'); insert into currency values('USD','no','no','American  
Dollar')
```

\*

ERROR at line 1:

ORA-00001: unique constraint (SYSTEM.SYS\_C007337) violated

```
SQL> insert into currency values('USD','no','no','US  
Dollar'); insert into currency values('USD','no','no','US  
Dollar')
```

\*

ERROR at line 1:

ORA-00001: unique constraint (SYSTEM.SYS\_C007337) violated

```
SQL> insert into currency_PROJ values('USD','no','no','US Dollar');
```

1 row created.

```
SQL> insert into currency_PROJ values('Euro','no','no','EURO');
```

1 row created.

18BIT0171  
18BIT0090

SQL> desc shares;

Name	Null?	Type
-----		
S_NAME	NOT NULL	VARCHAR2(20)
FACEVALUE		NUMBER(5,2)
NUMBER_OF_SHARES		NUMBER(5)

SQL> select \* from customer\_proj;

C_ID	PH_NO	NO_OF_SHARES_PURCHASED	NAME
-----			
111 8989898989		400	ABCDEF
222 8172856627		500	Sarthak
333 9889073505		600	Shivam
444 9140286432		700	Harshita
555 9794775121		200	Virat

SQL> insert into shares\_proj values('Shell ltd',300,40000,'Shell india',2);

1 row created.

SQL> insert into shares\_proj values('HPCL ltd',300,50000,'HPCL',5);

1 row created.

SQL> insert into shares\_proj values('bal ltd',100,300000,'balaji store',3);

insert into shares\_proj values('bal ltd',100,300000,'balaji store',3)



18BIT0171  
18BIT0090

\*

ERROR at line 1:

ORA-02291: integrity constraint (SYSTEM.SYS\_C007389) violated - parent key  
not found

SQL> insert into shares\_proj values('bal ltd',100,300000,'Balaji store',3);

1 row created.

SQL> insert into market\_lot\_proj values(20,'bal ltd');

1 row created.

SQL> insert into market\_lot\_proj values(40,'HPCL ltd');

1 row created.

SQL> insert into market\_lot\_proj values(100,'Shell ltd');

1 row created.

SQL> select \* from bill\_proj;

BILL_ID	BILL_CUST_ID	BILL_DESCN	BILL_DATE
1	111	Purchase	13-OCT-19

18BIT0171  
18BIT0090

2	222 Purchase	13-SEP-19
3	333 Sell	13-AUG-19
4	444 Purchase	13-JUL-19
5	333 Purchase	13-JUN-19

SQL> select \* from customer\_proj;

C_ID	PH_NO	NO_OF_SHARES_PURCHASED	NAME
111	8989898989	400	ABCDEF
222	8172856627	500	Sarthak
333	9889073505	600	Shivam
444	9140286432	700	Harshita
555	9794775121	200	Virat

SQL> insert into buy\_proj values('Shell ltd',222,295,315,to\_date('13-10-2019','dd-mm-yyyy'),to\_date('23-10-2019','dd-mm-yyyy'));

1 row created.

SQL> insert into buy\_proj values('HPCL',333,300,310,to\_date('03-10-2019','dd-mm-yyyy'),to\_date('13-10-2019','dd-mm-yyyy'));

insert into buy\_proj values('HPCL',333,300,310,to\_date('03-10-2019','dd-mm-yyyy'),to\_date('13-10-2019','dd-mm-yyyy'))

\*

ERROR at line 1:

ORA-02291: integrity constraint (SYSTEM.SYS\_C007402) violated - parent key

not found

**18BIT0171**  
**18BIT0090**

```
SQL> insert into buy_proj values('HPCL',333,300,310,to_date('03-10-2019','dd-  
mm- yyyy'),to_date('23-10-2019','dd-mm-yyyy'));
```

```
insert into buy_proj values('HPCL',333,300,310,to_date('03-10-2019','dd-mm-yyyy'),to_date('23-10-  
2019','dd-mm-yyyy'))
```

\*

ERROR at line 1:

ORA-02291: integrity constraint (SYSTEM.SYS\_C007402) violated - parent key

not found

SQL>

```
SQL> insert into buy_proj values('HPCL',333,300,310,to_date('03-10-2019','dd-  
mm- yyyy'),to_date('23-10-2019','dd-mm-yyyy'));
```

```
insert into buy_proj values('HPCL',333,300,310,to_date('03-10-2019','dd-mm-yyyy'),to_date('23-10-  
2019','dd-mm-yyyy'))
```

\*

ERROR at line 1:

ORA-02291: integrity constraint (SYSTEM.SYS\_C007402) violated - parent key

not found

```
SQL> insert into buy_proj values('HPCL ltd',333,300,310,to_date('03-10-2019','dd-  
mm- yyyy'),to_date('23-10-2019','dd-mm-yyyy'));
```

1 row created.

```
SQL> insert into brought_through_proj values('INR','Shell  
ltd'); insert into brought_through_proj values('INR','Shell ltd')
```

**18BIT0171**  
**18BIT0090**

\*

ERROR at line 1:

ORA-00942: table or view does not exist

SQL> insert into brought\_with\_proj values('INR','Shell ltd');

1 row created.

SQL> insert into brought\_with\_proj values('INR','HPCL ltd');

1 row created.

SQL> insert into brought\_with\_proj values('INR','bal ltd');

1 row created.

SQL> insert into sells\_through\_proj values(333,12346,to\_date('23-10-2019','dd-mm-yyyy'));

1 row created.

SQL> insert into sells\_through\_proj values(222,12347,to\_date('23-10-2019','dd-mm-yyyy'));

1 row created.

SQL> insert into managed\_by\_proj values(12346,'HPCL ltd');

18BIT0171  
18BIT0090

1 row created.

SQL> insert into managed\_by\_proj values(12347,'Shell ltd');

1 row created.

SQL> insert into managed\_by\_proj values(12348,'bal ltd');

1 row created.

SQL> insert into purchases\_through\_proj values(12346,333);

1 row created.

SQL> insert into purchases\_through\_proj values(12347,222);

1 row created.

SQL> select \* from company\_proj;

C_NAME	EXPENSES	REVENUE	REG_NO
--------	----------	---------	--------

-----

Reliance	5800000	7800000	11111
Shell india	4000000	7800000	22222
HPCL	7000000	8800000	33333
Balaji store	6000000	9800000	44444

**18BIT0171**  
**18BIT0090**

SQL> select \* from index\_proj;

REG\_NO DATE\_OF\_L

-----

11111 10-OCT-10

22222 10-SEP-10

33333 10-AUG-10

44444 10-JUL-10

55555 10-JUN-10

SQL> select \* from broker\_proj;

BROKER\_ID

-----

12345

12346

12347

12348

12349

SQL> select \* from currency\_proj;

CODE IS\_BA IS\_AC NAME

-----

INR Yes Yes Indian Rupees

USD no no US Dollar

Euro no no EURO

18BIT0171  
18BIT0090

SQL> select \* from shares\_proj;

S_NAME	FACE_VALUE	NO_OF_SHARES	C_NAME	BILL_ID
Reliance Ltd	450	40000	Reliance	1
Shell ltd	300	40000	Shell india	2
HPCL ltd	300	50000	HPCL	5
bal ltd	100	300000	Balaji store	3

SQL> select \* from market\_lot\_proj;

LOT_SIZE	S_NAME
10	Reliance Ltd
20	bal ltd
40	HPCL ltd
100	Shell ltd

SQL> select \* from customer\_proj;

C_ID	PH_NO	NO_OF_SHARES_PURCHASED	NAME
111	8989898989	400	ABCDEF
222	8172856627	500	Sarthak
333	9889073505	600	Shivam
444	9140286432	700	Harshita

18BIT0171  
18BIT0090

555 9794775121

200 Virat

SQL> select \* from bill\_proj;

BILL_ID	BILL_CUST_ID	BILL_DESCN	BILL_DATE
1	111	Purchase	13-OCT-19
2	222	Purchase	13-SEP-19
3	333	Sell	13-AUG-19
4	444	Purchase	13-JUL-19
5	333	Purchase	13-JUN-19

SQL> select \* from buy\_proj;

S_NAME	C_ID	PRICE_PURCHASED	PRICE_SOLD	DATE_OF_P	DATE_OF_S
Reliance Ltd	111	525	528	14-JUL-19	15-AUG-19
Shell ltd	222	295	315	13-OCT-19	23-OCT-19
HPCL ltd	333	300	310	03-OCT-19	23-OCT-19

SQL> select \* from brought\_with\_proj;

CODE S\_NAME

-- INR HPCL ltd

INR Reliance Ltd

INR Shell ltd



**18BIT0171**  
**18BIT0090**

18BIT0118 INR bal ltd

SQL> select \* from sells\_through\_proj;

C\_ID BROKER\_ID DATE\_OF\_S

-----

111 12345 15-OCT-19

333 12346 23-OCT-19

222 12347 23-OCT-19

SQL> select \* from managed\_by\_proj;

BROKER\_ID S\_NAME

-----

12345 Reliance Ltd

12346 HPCL ltd

12347 Shell ltd

12348 bal ltd

SQL> select \* from purchases\_through\_proj;

BROKER\_ID C\_ID

-----

12345 111

12346 333

12347 222

## 18BIT0171

## 18BIT0090

18BIT0118 SQL> commit;

Commit

complete. SQL>

```
Microsoft Windows [Version 10.0.17134.0]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\18BIT0090> sqlplus

SQL*Plus: Release 11.2.0.4.0 Production on Sat Nov 2 17:40:30 2018

Copyright (c) 1982, 2011, Oracle. All rights reserved.

Enter user-name: system
Enter password:
connected to:
Oracle Database 11g (Express Edition Release 11.2.0.4.0 - 64bit Production)

SQL> select * from company_pro;
 2 -

```

NAME	EXPENSE	REVENUE	APR_NO
Bullance	1000000	7000000	11111

```
SQL> insert into index_pro values(12222,to_date('10-09-2018','dd-mm-yyyy'));
insert into index_pro values(22222,to_date('10-09-2018','dd-mm-yyyy'))
ERROR at line 1:
ORA-00017: missing column

SQL> insert into index_pro values(22222,to_date('10-09-2018','dd-mm-yyyy'));
1 row created.

SQL> insert into index_pro values(11111,to_date('10-09-2018','dd-mm-yyyy'));
1 row created.

SQL> insert into index_pro values(44444,to_date('10-09-2018','dd-mm-yyyy'));
1 row created.

SQL> insert into index_pro values(12222,to_date('10-09-2018','dd-mm-yyyy'));
1 row created.

SQL> insert into company_pro values('Shell India',1000000,7000000,12222);
1 row created.

SQL> insert into company_pro values('Shell India',7000000,1000000,12222);
```

```
Microsoft Windows [Version 10.0.17134.0]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\18BIT0090> sqlplus

SQL> insert into bill_pro values(4,004,'Purchase',to_date('11-07-2018','dd-mm-yyyy'));
1 row created.

SQL> insert into bill_pro values(3,333,'Purchase',to_date('11-06-2018','dd-mm-yyyy'));
1 row created.

SQL> insert into broker_pro values(11346);
1 row created.

SQL> insert into broker_pro values(11347);
1 row created.

SQL> insert into broker_pro values(11348);
1 row created.

SQL> insert into broker_pro values(11349);
1 row created.

SQL> insert into currency values('USD','us','us','US Dollar');
insert into currency values('USD','us','us','US Dollar')
ERROR at line 1:
ORA-00001: unique constraint (SYS.TP,SPS_5802137) violated

SQL> insert into currency values('USD','us','us','US Dollar');
insert into currency values('USD','us','us','US Dollar')
ERROR at line 1:
ORA-00001: unique constraint (SYS.TP,SPS_5802137) violated

SQL> insert into currency_PRO values('USD','us','us','US Dollar');
1 row created.

SQL> insert into currency_PRO values('Euro','eu','eu','EUR');
1 row created.

SQL> desc shares;
Name                               Null?    Type
-----

```

**18BIT0171**  
**18BIT0090**

```

SQL> desc shares;
+-----+
| name |          | Null? | Type |
+-----+
| S_SHARES |          | YES | NUMBER(20) |
| ACCOUNT |          | YES | NUMBER(5,2) |
| NAME_OF_SHARES |          | YES | VARCHAR2(5) |
+-----+

SQL> select * from customer_prof;
+-----+
| C_ID | P1_M0 M0 OF_SSHARES_PUBM0SG0 SHRS |
+-----+
| 111 | 888997506 | 500 | APJal |
| 222 | 817295427 | 500 | Sarthak |
| 333 | 888977505 | 500 | Shivam |
| 444 | 914828643 | 700 | Harshita |
| 555 | 879477511 | 200 | Virat |
+-----+

SQL> insert into shares_prof values('Shell Ltd',100,50000,'India',2);
1 row created.

SQL> insert into shares_prof values('NPL Ltd',200,30000,'NPL',3);
1 row created.

SQL> insert into shares_prof values('hal ltd',100,100000,'halajl share',3);
insert into shares_prof values('hal ltd',100,50000,'halajl store',3);
*
ERROR at line 1:
ORA-02291: integrity constraint (SYSTEM.VTS_C097100) violated - parent key not found

SQL> insert into shares_prof values('hal ltd',200,100000,'halajl store',3);
1 row created.

SQL> insert into market_int_prof values(20,'hal ltd');
1 row created.

SQL> insert into market_int_prof values(40,'NPL ltd');
1 row created.

SQL> insert into market_int_prof values(100,'shell ltd');
1 row created.

-- Company Prof
SQL> insert into company_prof values('NP') , (100000,500000,XXXX);
1 row created.

SQL> insert into company_prof values('halajl share',500000,100000,AAAA);
1 row created.

SQL> insert into shares_prof values('Shell Ltd',200,30000,'shell India');
insert into shares_prof values('Shell Ltd',200,50000,'shell India');
*
ERROR at line 1:
ORA-00547: not enough values

SQL> insert into customer_prof values(222,817295427,500,'Sarthak');
1 row created.

SQL> insert into customer_prof values(333,888977505,500,'Shivam');
1 row created.

SQL> insert into customer_prof values(444,914828643,700,'Harshita');
1 row created.

SQL> insert into customer_prof values(555,879477511,200,'Virat');
1 row created.

-- Bill Prof
SQL> insert into bill_prof values(2,222,Purchase,to_date('11-06-2019','dd-mm-yyyy'));
insert into bill_prof values(3,222,Purchase,to_date('11-06-2019','dd-mm-yyyy'));
*
ERROR at line 1:
ORA-00001: column not allowed here

SQL> insert into bill_prof values(2,222,'Purchase',to_date('11-06-2019','dd-mm-yyyy'));
1 row created.

SQL> insert into bill_prof values(3,333,'sell',to_date('11-06-2019','dd-mm-yyyy'));
1 row created.

SQL> insert into bill_prof values(4,444,'Purchase',to_date('11-07-2019','dd-mm-yyyy'));

```

# 18BIT0171

# 18BIT0090

```

Command Prompt - admin

SQL> select * from bill_proj;

BILL_ID BILL_COST_ID BILL_DESCN BILL_DATE
-----
1 111 Purchase 11-OCT-19
2 222 Purchase 11-SEP-19
3 333 Sell 11-NOV-19
4 444 Purchase 11-NOV-19
5 555 Purchase 11-NOV-19

SQL> select * from customer_proj;

C_ID P_ID NO_OF_SHARES PURCHASED NAME
-----
111 888888888 400 ABCDEF
222 8122858623 500 GHIJKL
333 888887788 600 MNPQRS
444 8108286432 700 RSTUVW
555 8784775521 800 VWXYZ

SQL> insert into buy_proj values('Shell',111,222,200,to_date('11-10-2019','dd-mm-yyyy'),to_date('21-10-2019','dd-mm-yyyy'));
1 row created.

SQL> insert into buy_proj values('WPL',333,300,310,to_date('01-10-2019','dd-mm-yyyy'),to_date('11-10-2019','dd-mm-yyyy'));
insert into buy_proj values('WPL',333,300,310,to_date('01-10-2019','dd-mm-yyyy'),to_date('11-10-2019','dd-mm-yyyy'))
ERROR at line 1:
ORA-02291: integrity constraint (SYSTEM.SYS_C007402) violated - parent key not found

SQL> insert into buy_proj values('WPL',333,300,310,to_date('01-10-2019','dd-mm-yyyy'),to_date('21-10-2019','dd-mm-yyyy'));
insert into buy_proj values('WPL',333,300,310,to_date('01-10-2019','dd-mm-yyyy'),to_date('21-10-2019','dd-mm-yyyy'))
ERROR at line 1:
ORA-02291: integrity constraint (SYSTEM.SYS_C007402) violated - parent key not found

SQL> insert into buy_proj values('WPL',333,300,310,to_date('01-10-2019','dd-mm-yyyy'),to_date('21-10-2019','dd-mm-yyyy'));
insert into buy_proj values('WPL',333,300,310,to_date('01-10-2019','dd-mm-yyyy'),to_date('21-10-2019','dd-mm-yyyy'))
ERROR at line 1:
ORA-02291: integrity constraint (SYSTEM.SYS_C007402) violated - parent key not found

```

```

Command Prompt - admin

SQL> select * from company_proj;

C_NAME EXPENSES REVENUE P_ID NO
-----
Bellande 5000000 7000000 11111
Shell India 4000000 7000000 22222
WPL 3000000 6000000 33333
Relax store 6000000 8000000 44444

SQL> select * from index_proj;

REQ_NO DATE_OF_I
-----
11111 10-OCT-19
22222 10-SEP-19
33333 10-NOV-19
44444 10-NOV-19
55555 10-NOV-19

SQL> select * from broker_proj;

BROKER_ID
-----
12345
12346
12347
12348
12349

SQL> select * from currency_proj;

CURR ID BA IS AC NAME
-----
USD 00 no Yes Indian Rupee
EUR 00 no No US Dollar
GBP 00 no No British Pound

SQL> select * from shares_proj;

S_NAME FACE_VALUE NO_OF_SHARES C_NAME BILL_ID
-----
Bellande Ltd 250 40000 Bellande 1
Shell India 300 20000 Shell India 2
WPL 200 30000 WPL 3
Relax store 100 300000 Relax store 4

SQL> select * from market_lot_proj;

LOT_SIZE S_NAME

```

```
Command Prompt : sqlplus
SQL> select * from market_int_proj;

SQL> select * from customer_proj;

SQL> select * from bill_proj;

SQL> select * from buy_proj;

SQL> select * from brought_with_proj;

SQL> select * from sells_through_proj;
```

STOCK_SIZE	S_NAME
10	Reliance Ltd
20	Rel Ltd
40	WCL Ltd
100	Shell Ltd

C_ID	PH_NO	ID	OF	SQUARES	PURCHASED	SNAME
111	0090808080			400	ARCDET	
222	017205627			500	Serious	
333	0090877595			600	Shyam	
444	0100266433			700	Harshita	
555	0794775321			800	Virat	

BILL_ID	BILL_CUST_ID	BILL_REASON	BILL_DATE
1	111	Purchase	13-OCT-19
2	222	Purchase	13-SEP-19
3	333	Sell	13-AUG-19
4	444	Purchase	13-JUL-19
5	555	Purchase	13-JUN-19

S_NAME	C_ID	PRICE	PURCHASED	PRICE	SALE	DATE	OF	P	DATE	OF	S
Reliance Ltd	111	525	520	14-JUL-19	15-AUG-19						
Shell Ltd	222	295	315	13-OCT-19	23-OCT-19						
WCL Ltd	333	100	110	03-OCT-19	23-OCT-19						

STOCK	S_NAME
100	WCL Ltd
100	Reliance Ltd
100	Shell Ltd
100	Rel Ltd

C_ID	BOOKED_ID	DATE	OF	S
------	-----------	------	----	---

## Retrieval:

1. To find the share name given we have the date of listing

```
SQL> select s_name from shares_proj where c_name in(select c_name from  
company_proj where reg_no in(select reg_no from index_proj where date_of_listing =  
'10-OCT-10'));
```

S\_NAME

-----

Reliance Ltd

```
SQL> select s_name from shares_proj where c_name in(select c_name from company_proj where reg_no in(select reg_no from index_proj where date_of_listing = '10-OCT-10'));  
S_NAME  
-----  
Reliance ltd
```

- 2 To display only those shares where the face\_value>200 and to print them in order of their share name.

```
SQL> select c_name from company_proj minus select s_name from shares_proj where  
face_value>200 group by s_name;
```

C\_NAME

-----

Balaji store

HPCL

Reliance

Shell india

1. .

```
SQL> select c_name from company_proj minus select s_name from shares_proj where face_value>500;

C_NAME
-----
Salaji store
HPCL
Bullrunn
Shell India
```

2. To find out the share name and name of the customer given the number of shares\_purchased>500.

```
SQL> select s_name,name from shares_proj inner join bill_proj on
bill_proj.bill_id=shares_proj.bill_id inner join customer_proj on
customer_proj.c_id=bill_proj.bill_cust_id where
customer_proj.no_of_shares_purchased>500;
```

S_NAME	NAME
-----	
HPCL ltd	Shivam
bal ltd	Shivam

```
SQL> select s_name,name from shares_proj inner join bill_proj on bill_proj.bill_id=shares_proj.bill_id inner join customer_proj on customer_proj.c_id=bill_proj.bill_cust_id where customer_proj.no_of_shares_purchased>500;

S_NAME      NAME
-----
HPCL Ltd    Shivam
bal ltd     Shivam
```

3. To display all the shares name which starts with 'H' or 'S' and display them in order of their share name.

SQL> select s.s\_name from shares\_proj s,market\_lot\_proj m where m.s\_name=s.s\_name  
group by s.s\_name having (s.s\_name like 'H%' or s.s\_name like 'S%');

S\_NAME

-----

HPCL ltd

Shell ltd

```
SQL> select s.s_name from shares_proj s,market_lot_proj m where m.s_name=s.s_name group by s.s_name having (s.s_name like 'H%' or s.s_name like 'S%');  
S_NAME  
-----  
HPCL ltd  
Shell ltd  
SQL>
```



## **Updation:**

1. To update the customers phone number(interactively) based on the customer id provided to them.

```
SQL> update customer_proj set ph_no=&ph_no where c_id in(select c_id from  
customer_proj where c_id=&c_id);
```

Enter value for ph\_no: 8888888888 Enter

value for c\_id: 111

old 1: update customer\_proj set ph\_no=&ph\_no where c\_id in(select c\_id from  
customer\_proj where c\_id=&c\_id)

new 1: update customer\_proj set ph\_no=8888888888 where c\_id in(select c\_id from  
customer\_proj where c\_id=111)

1 row updated.

```
SQL> select * from customer_proj
```

2 ;

C\_ID PH\_NO NO\_OF\_SHARES\_PURCHASED NAME

111	8888888888	400	ABCDEF
222	8172856627	500	Sarthak
333	9889073505	600	Shivam
444	9140286432	700	Harshita
555	9794775121	200	Virat

```
SQL> update customer_proj set ph_no=&ph_no where c_id in(select c_id from customer_proj where c_id=&c_id);
Enter value for ph_no: 8888888888
Enter value for c_id: 111
old 1: update customer_proj set ph_no=&ph_no where c_id in(select c_id from customer_proj where c_id=&c_id)
new 1: update customer_proj set ph_no=8888888888 where c_id in(select c_id from customer_proj where c_id=111)

1 row updated.

SQL> select * from customer_proj
2 ;
```

C_ID	PH_NO	NO_OF_SHARES_PURCHASED	NAME
111	8888888888	400	ABCDEF
222	8172856627	500	Sarthak
333	9889073505	600	Shivam
444	9140286432	700	Harshita
555	9794775121	200	Virat

```
SQL>
```

- To update the lot size of a particular share(interactively) based on the share name provided.

```
SQL> update market_lot_proj set lot_size=&lot_size where s_name in(select
s_name from shares_proj where s_name='&s_name');
```

Enter value for lot\_size: 20

Enter value for s\_name: Reliance Ltd

old 1: update market\_lot\_proj set lot\_size=&lot\_size where s\_name in(select s\_name from shares\_proj where s\_name='&s\_name')

new 1: update market\_lot\_proj set lot\_size=20 where s\_name in(select s\_name from shares\_proj where s\_name='Reliance Ltd')

1 row updated.

SQL> select \* from market\_lot\_proj 2 ;

LOT_SIZE	S_NAME
----------	--------

-----

20	Reliance Ltd
----	--------------

20	bal ltd
----	---------

40	HPCL ltd
----	----------

100	Shell ltd
-----	-----------

```
SQL> update market_lot_proj set lot_size=&lot_size where s_name in(select s_name from shares_proj where s_name='&s_name'
);
Enter value for lot_size: 20
Enter value for s_name: Reliance Ltd
old 1: update market_lot_proj set lot_size=&lot_size where s_name in(select s_name from shares_proj where s_name='&s_n
ame')
new 1: update market_lot_proj set lot_size=20 where s_name in(select s_name from shares_proj where s_name='Reliance Lt
d')
1 row updated.
SQL> select * from market_lot_proj
2 ;

 LOT_SIZE S_NAME
-----
      20 Reliance Ltd
      20 bal ltd
      40 HPCL ltd
     100 Shell ltd
SQL>
```

### **Deletion:**

1. To delete a particular share name from database where share name is taken as input from user interactively.

SQL> delete from market\_lot\_proj where s\_name in(select s\_name from shares\_proj where s\_name='&s\_name');

Enter value for s\_name: Reliance

old 1: delete from market\_lot\_proj where s\_name in(select s\_name from shares\_proj where s\_name='&s\_name')

new 1: delete from market\_lot\_proj where s\_name in(select s\_name from shares\_proj where s\_name='Reliance')

0 rows deleted.

```
SQL> delete from market_lot_proj where s_name in(select s_name from shares_proj where s_name='&s_name');
Enter value for s_name: Reliance
old 1: delete from market_lot_proj where s_name in(select s_name from shares_proj where s_name='&s_name')
new 1: delete from market_lot_proj where s_name in(select s_name from shares_proj where s_name='Reliance')

0 rows deleted.

SQL>
```

2. To delete a particular currency code from the database where code starts with 'f'.

SQL> delete from currency\_proj where code in(select code from currency\_proj where code like 'f%');

0 rows deleted.

```
SQL> delete from currency_proj where code in(select code from currency_proj where code like 'f%');

0 rows deleted.

SQL>
```

18BIT0171  
18BIT0090