REPORT

----Report for Profit gained---------------

create view PROFIT\_GAINED as

select p.Trans\_id as Transaction\_ID,

p.Stock\_name as Stock\_Name,

p.Ticker\_symbol as Stock\_ID,

p.client\_id as Client\_ID,

p.noofstocks as No\_of\_stocks,

p.profit\_amount as Profit\_gained,

c.category\_name as Stock\_Category\_name

from profit p

inner join stock s

on s.ticker\_symbol=p.ticker\_symbol

inner join category c

on c.cat\_id=s.cat\_id;

select \* from profit\_gained;

select \* from CLIENT\_stock\_match;

select \* from user\_tables where table\_name like '%CLIENT%';

--------Report for proposal sent to the client

create view proposal\_sent as

select CS.client\_id as client\_id,

CS.stock\_name,

C.category\_name,

CASE when CS.PARAMETER\_VALUE >= 90 THEN 'EXTREMELY HIGH'

when CS.PARAMETER\_VALUE >= 80 THEN 'HIGH'

when CS.PARAMETER\_VALUE >= 70 THEN 'MODERATE'

when CS.PARAMETER\_VALUE >= 60 THEN 'LOW'

ELSE 'EXTREMELY LOW'

END AS STOCK\_PERFORMANCE

from client\_stock\_match CS

inner join CATEGORY C

ON CS.CAT\_ID = C.CAT\_ID;

-----accept/reject proposal status

create view status\_report as

select p.trans\_id as transaction\_id,

p.client\_id,

s.stock\_name,

To\_char(p.trans\_date,'MONTH') as transaction\_month,

p.status as Client\_status

from proposal\_status p

inner join stock s

on s.ticker\_symbol=p.ticker\_symbol

order by p.status, p.trans\_date ;

select \* from status\_report;

------------------------------------------Create Table---------------------------

Create table Client\_(

Client\_Id int PRIMARY KEY,

First\_Name varchar(255) NOT NULL,

Last\_Name varchar(255) NOT NULL,

Street varchar(255) NOT NULL,

Contact\_Number varchar(10) NOT NULL,

SSN varchar(10) NOT NULL, -----UNIQUE---

Email varchar(50) NOT NULL,

City varchar(50) NOT NULL,

State varchar(50) NOT NULL,

Zip\_code varchar(25) NOT NULL) ----NUMBER(5)-----

---constraint CHECKEMAIL CHECK (Email like '%@%.com%')--);

Create table Client\_strat(

Strategy\_ID varchar(5) NOT NULL,

Cat\_ID varchar(25) NOT NULL,

Brand\_value varchar(25) NOT NULL,

Investment varchar(25) NOT NULL,

Parameter\_Value int NOT NULL,

PRIMARY KEY (Strategy\_ID)

);

create table STOCK (

Stock\_Name varchar(20),

Ticker\_symbol varchar(10) NOT NULL PRIMARY KEY,

Cat\_Id varchar(15) NOT NULL,

Risk varchar(10),

Brand\_value varchar(10),

Investment varchar(10),

current\_value int NOT NULL);

create table Category(

Cat\_ID varchar(25) NOT NULL PRIMARY KEY,

Category\_Name varchar(25) NOT NULL,

Brokerage\_Rate int NOT NULL);

create table Stock\_Parameter(

Ticker\_Symbol varchar(10) PRIMARY KEY,

Cat\_ID varchar(10) NOT NULL,

News int NOT NULL,

Dividends\_Announced int NOT NULL,

Product\_Release int NOT NULL,

New\_Contract int NOT NULL,

Employee\_Layoff int NOT NULL,

Takeover int NOT NULL,

Mgmt\_Changes int NOT NULL,

Scandals int NOT NULL,

EPS int NOT NULL,

PE\_ratio int NOT NULL,

Exchange\_rate int NOT NULL);

create table client\_portf(

Client\_Id varchar(10) NOT NULL,

Category\_Name varchar(25) NOT NULL,

Cat\_ID Varchar(10) NOT NULL

);

create table Client\_Category (

Client\_ID int NOT NULL ,

Cat\_ID Varchar(25) NOT NULL ,

Brokerage\_Rate int NOT NULL);

create table Strategy\_Category (

Client\_ID VARCHAR(10) NOT NULL ,

Cat\_ID Varchar(10) NOT NULL ,

Strategy\_ID Varchar(10) NOT NULL ,

Brand\_Value Varchar(25) NOT NULL ,

Investment Varchar(25) NOT NULL ,

Parameter\_Value int NOT NULL );

create table PROPOSAL\_STATUS (

TRANS\_ID VARCHAR(10) PRIMARY KEY,

MATCH\_ID VARCHAR(3),

client\_id VARCHAR(5),

Ticker\_symbol VARCHAR(3),

NOOFSTOCKS NUMBER(4) default 0 ,

STATUS VARCHAR(6),

TRANS\_DATE DATE DEFAULT SYSDATE,

CONSTRAINT CHECK\_STATUS CHECK (STATUS IN ('ACCEPT','REJECT') AND ((STATUS = 'ACCEPT' and NOOFSTOCKS > 0) OR (STATUS = 'REJECT' and NOOFSTOCKS = 0)))

);

-------------------------------------------Create table using select ------------------------------------

create table STOCK\_DECISION as (SELECT Rownum as Dec\_ID, e.parameter\_value,S.\* FROM STOCK S

inner join evaluation E on E.ticker\_symbol = S.ticker\_symbol) ;

create table evaluation as

(select Ticker\_Symbol, (News+Dividends\_Announced+Product\_Release+New\_Contract+Employee\_Layoff+Takeover+Mgmt\_Changes+Scandals+EPS+PE\_Ratio+Exchange\_Rate)as Parameter\_Value

from Stock\_Parameter);

create table client\_stock\_match AS

select concat('M0',rownum) as Match\_ID,S.\*,C.Client\_ID, St.Strategy\_ID from STOCK\_DECISION S

inner join Client\_Category C

on S.Cat\_ID = C.Cat\_ID

inner join Strategy\_Category St

on C.Client\_ID = St.Client\_ID

and C.Cat\_ID = St.Cat\_ID

and S.Brand\_Value = St.Brand\_Value

and S.Investment = St.Investment

and S.Parameter\_Value >= St.Parameter\_Value

order by Match\_ID,C.Cat\_ID;

Create table Client\_Stock\_Proposal as

select CC.Brokerage\_Rate, CS.MATCH\_ID, CS.Stock\_Name, CS.Ticker\_symbol, CS.Cat\_Id, CS.Current\_value, CS.Client\_Id

from client\_stock\_match CS inner join Client\_Category CC

on CC.Client\_Id = CS.Client\_Id and CS.Cat\_Id = CC.Cat\_Id;

create table TRANS\_HEADER AS(

select p.TRANS\_ID,c.client\_id,c.Ticker\_Symbol,c.Stock\_Name,c.Cat\_ID

from Client\_Stock\_Proposal c

inner join PROPOSAL\_STATUS p

on p.MATCH\_ID=c.MATCH\_ID

and p.STATUS='ACCEPT');

create table TRANS\_DETAIL AS

select p.TRANS\_ID,c.Ticker\_Symbol,c.Stock\_Name,c.Cat\_ID,p.NOOFSTOCKS,c.current\_value,c.brokerage\_rate,p.TRANS\_DATE

from Client\_Stock\_Proposal c

inner join PROPOSAL\_STATUS p

on p.MATCH\_ID=c.MATCH\_ID

inner join TRANS\_HEADER t

on t.client\_id = c.client\_id

and p.STATUS='ACCEPT';

create table TRANS\_DELETION AS

select p.TRANS\_ID,p.MATCH\_ID,c.client\_id,c.Ticker\_Symbol,c.Stock\_Name,p.TRANS\_DATE

from Client\_Stock\_Proposal c

inner join PROPOSAL\_STATUS p

on p.MATCH\_ID = c.MATCH\_ID

and p.STATUS='REJECT';

create table PROFIT as

select concat('P0',rownum) as Profit\_ID,c.client\_id,c.Ticker\_Symbol,c.Stock\_Name,p.TRANS\_ID,c.Brokerage\_rate,p.noofstocks,c.Brokerage\_rate\*p.noofstocks as Profit\_Amount

from Client\_Stock\_Proposal c

inner join PROPOSAL\_STATUS p

on p.MATCH\_ID = c.MATCH\_ID ;

----------------------------------------------------------------------------