A

Project Report

on

ONLINE RECHARGE SYSTEM

Developed by

DHARAM V. SHAH -Department of IT, DD University

KULDEEP R. SAVALIYA-Department of IT, DD University

Guided By

PROF. Sunil K.Vithlani

Department of Information Technology

Faculty of Technology

DD University

Department of Information Technology

Faculty of Technology, Dharmsinh Desai University

College Road, Nadiad-387001

October - 2018

TABLE OF CONTENTS

I. Certificate I

II. Acknowledgement II

1. SYSTEM OVERVIEW 4

1.1 Current system
1.2 Objectives of the Proposed System
1.3 Advantages of the Proposed system (over current)

2. E-R DIAGRAM 5

2.1 Entities…………………………………………………………………..5
2.2 Relationships………………………………………………………………6
2.3 Mapping Constraints………………………………………………………7

3. DATA DICTIONARY 8

4. SCHEMA DIAGRAM 13

5. DATABASE IMPLEMENTATION

5.1 Create Schema 14
5.2 Insert Data values 16
5.3 Queries (Based on functions, group by, having, joins, sub query etc.) 18
5.4 PL/SQL Blocks (Procedures and Functions) 23
5.5 Views 26
5.6 Triggers 27
5.7 Cursors. 28

6. FUTURE ENHANCEMENTS OF THE SYSTEM 32

7. BIBLIOGRAPHY 33

DHARMSINH DESAI UNIVERSITY

NADIAD-387001, GUJARAT

CERTIFICATE

This is to certify that the project entitled "ONLINE RECHARGE SYSTEM" is a bonafide report of the work carried out by

1) Mr. Dharam V. Shah **Student ID No:**16ITUOS077

2) Mr. Kuldeep R.Savaliya **Student ID No:**16ITUOS100

of Department of Information Technology, semester V, under the guidance and supervision for the subject Database Management System. They were involved in Project training during academic year 2018-2019..

Prof. Sunil K. Vithlani
(Project Guide)
Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad
Date:

Prof. Vipul Dabhi
Head , Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad
Date:

ACKNOWLEDGEMENT

In the accomplishment of this project successfully, my team would like to thank all the people concerned with this project who has given me the support.

I would like to thank my *Prof. Sunil sir* whose valuable guidance has been the ones that helped my team patch this project and make it full proof success. His suggestions and instructions has served as the major contributor towards the completion of project.

Thanking You,
Yours Sincerely,

Dharam Shah (IT-109)
Kuldeep Savaliya (IT-106)

System overview

1.1 Current system:

Now a days most of the people around the world are using mobile phones for communication.For this they require to recharge frequently and sometimes in emergency they want to recharge very urgently.At this time they don’t have the time to go to any shop and Do their recharge and so for this before their recharge ends they can do it through this application of online recharging which will be beneficial.

1.2 Objectives of Proposed System:

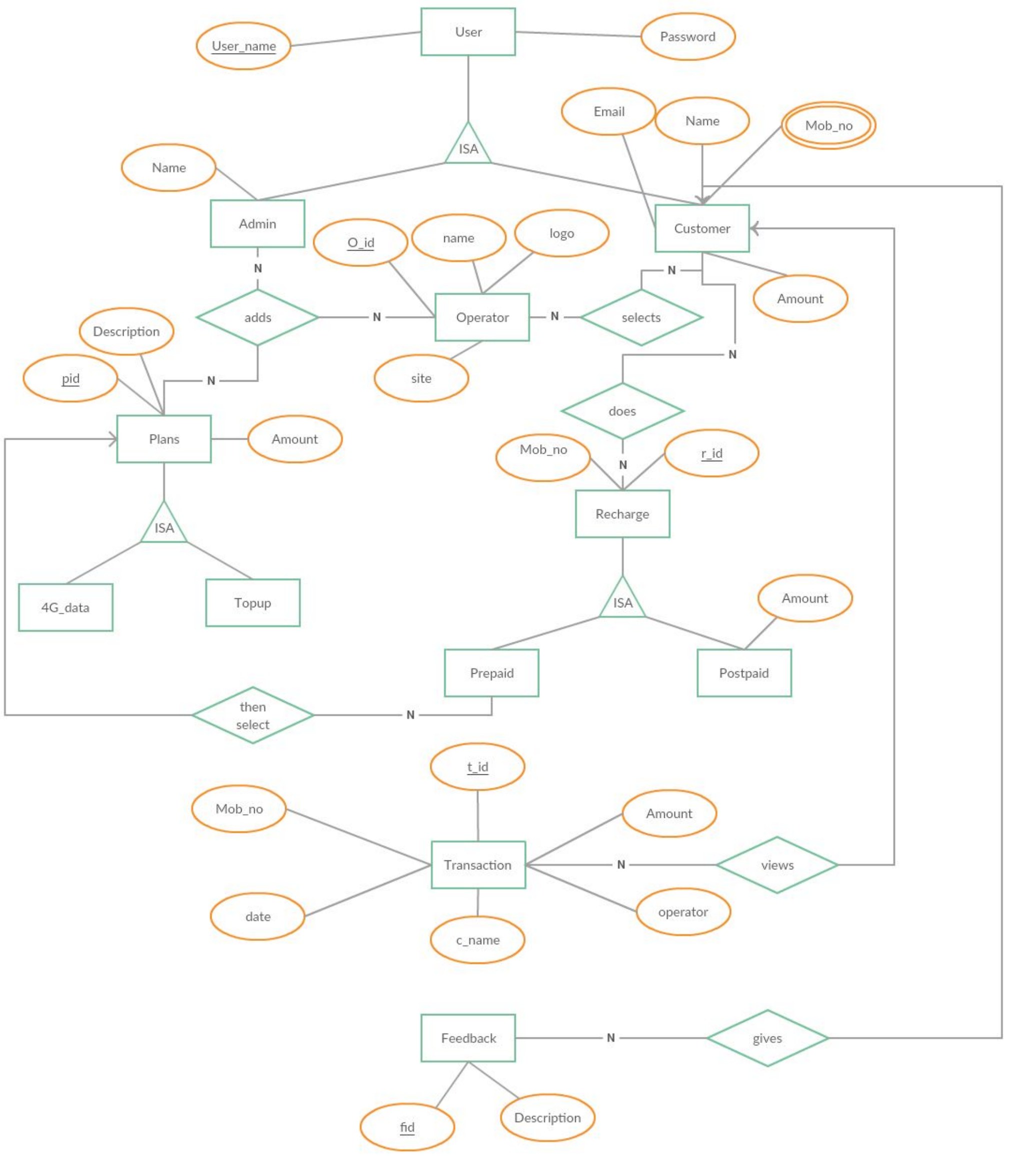
This is a project based on PL/SQL language. In this, a customer can recharge online on website. Every website maintains its own database system which provides all the facilities. Customer can retrieve data of his choice easily. Then the total amount that has to be paid is calculated automatically. The Customer will be shown his transaction history with time and date.And also it contains many procedures to manage database. All updates done on data items is reflected on database.

1.3 Advantages of Proposed System:

The main advantage of this system is that the customer can do recharge instantly at their home .For all the mobile numbers that he has their recharge can be done through this system he does not need to go to the service provider.

Transactions are secured and the statement of this is also provided. By chance if the transaction fails the money is guaranteed back. User friendly system for admin to add and delete tarrif.

ER DIAGRAM


User
User_name
Password
ISA
Email
Name
Mob_no
Name
Admin
Customer
O_id
name
logo
N
adds
N
Operator
N
selects
N
Amount
Description
pid
N
site
N
does
Plans
Amount
Mob_no
r_id
N
Recharge
ISA
ISA
4G_data
Topup
Amount
Prepaid
Postpaid
then select
N
t_id
Mob_no
Amount
Transaction
N
views
date
operator
c_name
Feedback
N
gives
fid
Description

2.1 ENTITIES

- USER
- ADMIN
- CUSTOMER
- OPERATOR
- PLANS
- RECHARGE
- FEEDBACK
- TRANSACTION_HISTORY
- PREPAID
- POSTPAID
- DATA
- TOPUP
- Mobile_no

2.2 Relationships

- Selects
- Does
- Adds
- Views
- Gives

- Then select

2.3 Mapping Constraints

- Selects - Many to Many
- Adds - Many to Many
- Does - Many to Many
- Views - Many to One
- Gives - Many to One
- Then Selects - Many to One

DATA DICTIONARY

adds

Column	Type	Null	Default	Links to
username *(Primary)*	varchar(15)	No		admin -> username
oid *(Primary)*	int(11)	No		operator -> oid
pid *(Primary)*	int(11)	No		plans -> pid

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	oid	2	A	No	
				pid	2	A	No	
				username	2	A	No	
pid	BTREE	No	No	pid	2	A	No	
username	BTREE	No	No	username	2	A	No	

admin

Column	Type	Null	Default	Links to
username *(Primary)*	varchar(20)	No		
password	varchar(15)	Yes	*NULL*	
name	char(20)	Yes	*NULL*	

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	username	16	A	No	

customer

Column	Type	Null	Default	Links to
username *(Primary)*	varchar(15)	No		
password	varchar(15)	No		

email	varchar(25)	No		
name	char(20)	No		
Mob_no	bigint(10)	No		
oid	int(11)	Yes	*NULL*	operator -> oid

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	username	1	A	No	
oid	BTREE	No	No	oid	1	A	Yes	

data

Column	Type	Null	Default	Links to
pid *(Primary)*	int(11)	No		plans -> pid

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	pid	0	A	No	

does

Column	Type	Null	Default	Links to
username *(Primary)*	varchar(15)	No		customer -> username
rid *(Primary)*	int(11)	No		recharge -> rid

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	username	0	A	No	
				rid	0	A	No	
rid	BTREE	No	No	rid	0	A	No	

feedback

Column	Type	Null	Default	Links to
fid *(Primary)*	int(11)	No		
username	varchar(15)	No		customer -> username
Description	mediumtext	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	fid	0	A	No	
username	BTREE	No	No	username	0	A	No	

mobile_no

Column	Type	Null	Default	Links to
username *(Primary)*	varchar(15)	No		customer -> username
Mob_1	bigint(10)	No		
Mob_2	bigint(10)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	username	0	A	No	

operator

Column	Type	Null	Default	Links to
oid *(Primary)*	int(11)	No		
name	varchar(15)	Yes	*NULL*	
site	varchar(20)	Yes	*NULL*	
logo	varchar(15)	Yes	*NULL*	

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	oid	10	A	No	

plans

Column	Type	Null	Default	Links to	Comments	MIME
pid *(Primary)*	int(11)	No				
Description	mediumtext	No				
Amount	decimal(6,2)	No				

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	pid	2	A	No	

postpaid

Column	Type	Null	Default	Links to	Comments	MIME
rid *(Primary)*	int(11)	No		recharge -> rid		
Amount	decimal(6,2)	No				

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	rid	0	A	No	

prepaid

Column	Type	Null	Default	Links to	Comments	MIME
pid *(Primary)*	int(11)	No		plans -> pid		
rid *(Primary)*	int(11)	No		recharge -> rid		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	pid	0	A	No	
				rid	0	A	No	
rid	BTREE	No	No	rid	0	A	No	

recharge

Column	Type	Null	Default	Links to	Comments	MIME
rid *(Primary)*	int(11)	No				
Mob_no	bigint(10)	No				

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	rid	0	A	No	

topup

Column	Type	Null	Default	Links to	Comments	MIME
pid *(Primary)*	int(11)	No		plans -> pid		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	pid	0	A	No	

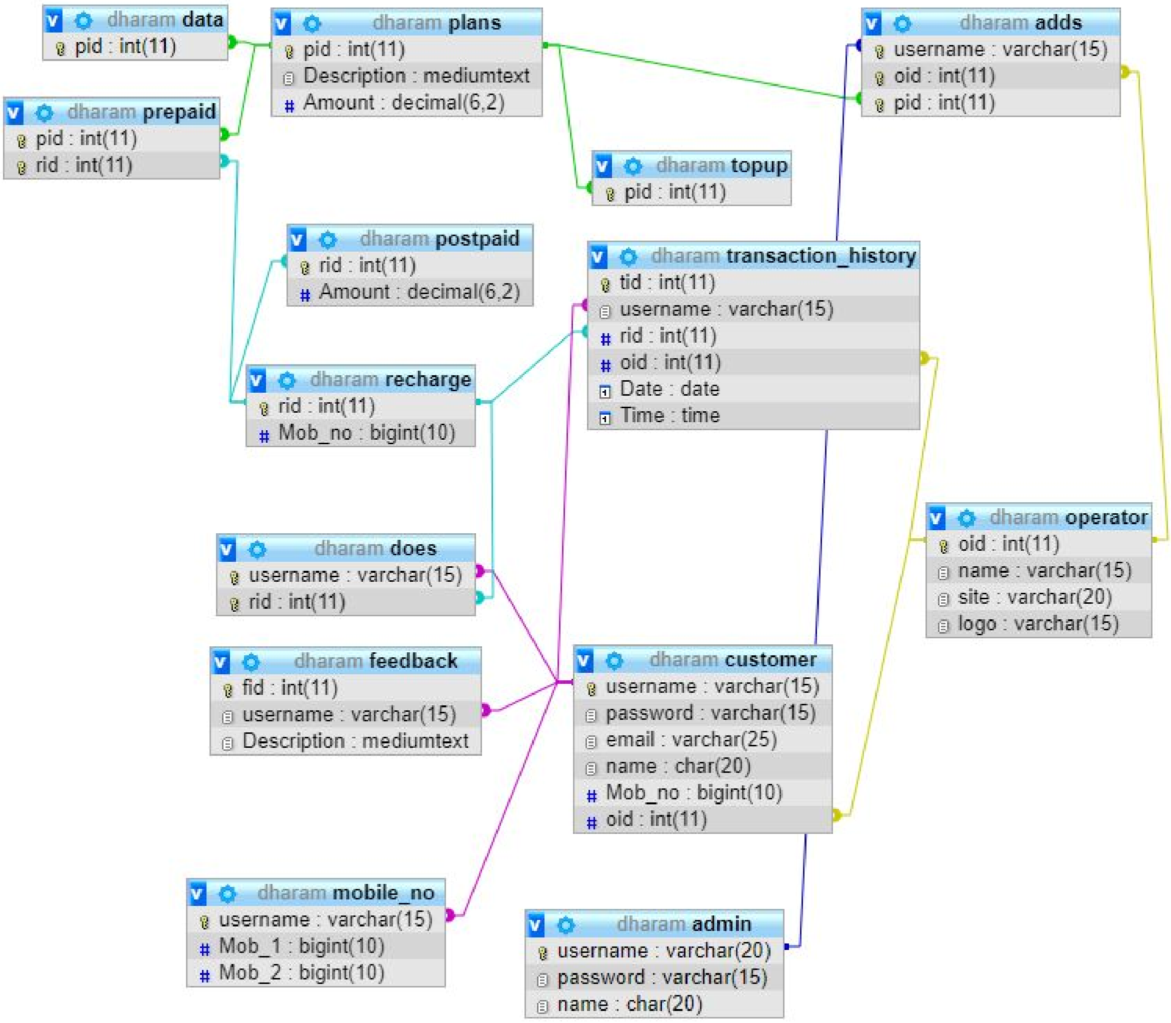
transaction_history

Column	Type	Null	Default	Links to
tid *(Primary)*	int(11)	No		
username	varchar(15)	No		customer -> username
rid	int(11)	No		recharge -> rid
oid	int(11)	No		operator -> oid
Date	date	No		
Time	time	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	tid	0	A	No	
rid	BTREE	No	No	rid	0	A	No	
oid	BTREE	No	No	oid	0	A	No	
username	BTREE	No	No	username	0	A	No	

ER SCHEMA


dharam data
pid : int(11)
dharam plans
pid : int(11)
Description : mediumtext
Amount : decimal(6,2)
dharam adds
username : varchar(15)
oid : int(11)
pid : int(11)
dharam prepaid
pid : int(11)
rid : int(11)
dharam topup
pid : int(11)
dharam postpaid
rid : int(11)
Amount : decimal(6,2)
dharam transaction_history
tid : int(11)
username : varchar(15)
rid : int(11)
oid : int(11)
Date : date
Time : time
dharam recharge
rid : int(11)
Mob_no : bigint(10)
dharam operator
oid : int(11)
name : varchar(15)
site : varchar(20)
logo : varchar(15)
dharam does
username : varchar(15)
rid : int(11)
dharam feedback
fid : int(11)
username : varchar(15)
Description : mediumtext
dharam customer
username : varchar(15)
password : varchar(15)
email : varchar(25)
name : char(20)
Mob_no : bigint(10)
oid : int(11)
dharam mobile_no
username : varchar(15)
Mob_1 : bigint(10)
Mob_2 : bigint(10)
dharam admin
username : varchar(20)
password : varchar(15)
name : char(20)

DATABASE IMPLEMENTATION

5.1 CREATE SCHEMAS

CREATE TABLE QUERIES

CREATE QUERY:

```
CREATE TABLE `admin` (
 `username` varchar(20) NOT NULL,
 `password` varchar(15) DEFAULT NULL,
 `name` char(20) DEFAULT NULL,
 PRIMARY KEY (`username`))
```

CREATE QUERY:

```
CREATE TABLE `customer` (
 `username` varchar(15) NOT NULL,
 `password` varchar(15) NOT NULL,
 `email` varchar(25) NOT NULL,
 `name` char(20) NOT NULL,
 `Mob_no` bigint(10) NOT NULL,
 `oid` int(11) DEFAULT NULL,
 PRIMARY KEY (`username`),
  FOREIGN KEY (`oid`) REFERENCES `operator` (`oid`)
)
```

CREATE QUERY:

```
CREATE TABLE `operator` (
 `oid` int(11) NOT NULL AUTO_INCREMENT,
 `name` varchar(15) DEFAULT NULL,
 `site` varchar(20) DEFAULT NULL,
 `logo` varchar(15) DEFAULT NULL,
 PRIMARY KEY (`oid`))
```

CREATE QUERY:

```
CREATE TABLE `transaction_history` (
 `tid` int(11) NOT NULL AUTO_INCREMENT,
 `username` varchar(15) NOT NULL,
 `rid` int(11) NOT NULL,
 `oid` int(11) NOT NULL,
 `Date` date NOT NULL,
 `Time` time NOT NULL,
 PRIMARY KEY (`tid`),
  FOREIGN KEY (`rid`) REFERENCES `recharge` (`rid`),
  FOREIGN KEY (`oid`) REFERENCES `operator` (`oid),
 FOREIGN KEY (`username`) REFERENCES `customer` (`username`)
)
```

CREATE QUERY:

```
CREATE TABLE `feedback` (
 `fid` int(11) NOT NULL AUTO_INCREMENT,
 `username` varchar(15) NOT NULL,
 `Description` mediumtext NOT NULL,
 PRIMARY KEY (`fid`),
 FOREIGN KEY (`username`) REFERENCES `customer` (`username`)
)
```

CREATE QUERY:

```
CREATE TABLE `mobile_no` (
 `username` varchar(15) NOT NULL,
 `Mob_1` bigint(10) NOT NULL,
 `Mob_2` bigint(10) NOT NULL,
 PRIMARY KEY (`username`),
  FOREIGN KEY (`username`) REFERENCES `customer` (`username`)
)
```

CREATE QUERY:

```
CREATE TABLE `does` (
 `username` varchar(15) NOT NULL,
 `rid` int(11) NOT NULL,
 PRIMARY KEY (`username`,`rid`),
 FOREIGN KEY (`username`) REFERENCES `customer` (`username`),
 FOREIGN KEY (`rid`) REFERENCES `recharge` (`rid`)
)
```

CREATE QUERY:

```
CREATE TABLE `postpaid` (
 `rid` int(11) NOT NULL,
 `Amount` decimal(6,2) NOT NULL,
 PRIMARY KEY (`rid`),
 FOREIGN KEY (`rid`) REFERENCES `recharge` (`rid`)
)
```

CREATE QUERY:

```
CREATE TABLE `prepaid` (
 `pid` int(11) NOT NULL,
 `rid` int(11) NOT NULL,
 PRIMARY KEY (`pid`,`rid`),
 FOREIGN KEY (`rid`) REFERENCES `recharge` (`rid`),
 FOREIGN KEY (`pid`) REFERENCES `plans` (`pid`)
)
```

CREATE QUERY:

```
CREATE TABLE `recharge` (
 `rid` int(11) NOT NULL AUTO_INCREMENT,
 `Mob_no` bigint(10) NOT NULL,
 PRIMARY KEY (`rid`)
)
```

CREATE QUERY:

```
CREATE TABLE `topup` (
 `pid` int(11) NOT NULL,
 PRIMARY KEY (`pid`),
 FOREIGN KEY (`pid`) REFERENCES `plans` (`pid`)
)
```

CREATE QUERY:

```
CREATE TABLE `plans` (
 `pid` int(11) NOT NULL AUTO_INCREMENT,
 `Description` mediumtext NOT NULL,
 `Amount` decimal(6,2) NOT NULL,
 PRIMARY KEY (`pid`)
)
```

5.2 INSERT DATA VALUES

Insert data queries

INSERT QUERY:

```
INSERT INTO
 `admin` (`username`, `password`, `name`) VALUES
('ahuggina', '2OdtokzyC1kC', 'Ardelia Huggin'),
('ctrimmingc', 'Grd95nEwoc', 'Cornall Trimming'),
('kmangeonb', 'PBvxerVw1XjJ', 'Kym Mangeon'),
```

INSERT QUERY:

```
INSERT INTO
```

```
`customer` (`username`, `password`, `email`, `name`, `Mob_no`, `oid`) VALUES
('dhas2332', 'dlssjdf', 'dahram777@gmail.com', 'asdffa', 5685686578, 1),
('jainish45', 'jai3434', 'jainishah@gmail.com', 'jainishah', 8585896491, 8),
```

INSERT QUERY:

```
INSERT INTO
`feedback` ( `username`, `Description`) VALUES
('dharam777', 'THIS WAS NICE RECHARGE'),
( 'aniket34', 'IT was really nice experience')
```

INSERT QUERY:

```
INSERT INTO
`operator` (`oid`, `name`, `site`, `logo`) VALUES
( 'Bsnl', 'www.bsnl.co.in', 'bsnl.png'),
( 'uninor', 'www.uninor.com', 'uninor.png')
```

INSERT QUERY:

```
INSERT INTO
`transaction_history` ( `username`, `rid`, `oid`, `Date`, `Time`) VALUES
( 'dharam777', 1, 1, '2018-09-05', '02:05:12'),
( 'aniket34', 2, 5, '2018-09-25', '12:11:03'),
( 'dhas2332', 3, 5, '2018-09-02', '09:17:02')
```

INSERT QUERY:

```
INSERT INTO
`plans` (`pid`, `Description`, `Amount`) VALUES
(1, '4G DATA 1GB', '36.00'),
(2, 'TOPUP FOR 200 RS', '200.00'),
(3, '4G DATA 2GB/DAY FOR 1 MONTH', '499.00')
```

INSERT QUERY:

```
INSERT INTO
`postpaid` (`rid`, `Amount`) VALUES
(1, '77.00'),
(2, '850.95')
```

INSERT QUERY:

```
INSERT INTO
`prepaid` (`pid`, `rid`) VALUES
(4, 4),
(6, 3)
```

INSERT QUERY:

```
INSERT INTO
`mobile_no` (`username`, `Mob_1`, `Mob_2`) VALUES
```

('jainish45', 7985220256, 9825205653),
('jas789', 9429352869, NULL)

5.3 QUERIES

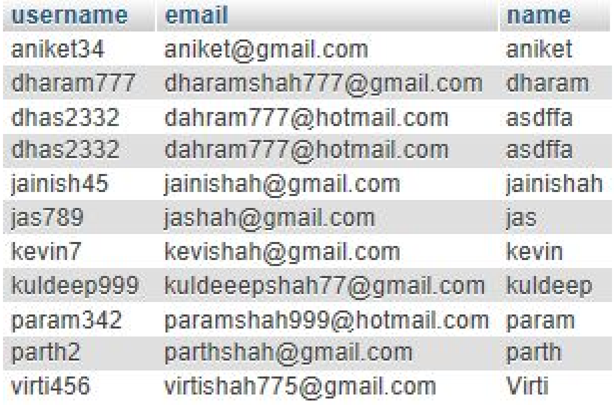
1)Display the all customers detail who did recharge ?

QUERY:

SELECT c.username,c.email,c.name from does d

JOIN customer c on c.username=d.username;

OUTPUT:



username	email	name
aniket34	aniket@gmail.com	aniket
dharam777	dharamshah777@gmail.com	dharam
dhas2332	dahram777@hotmail.com	asdffa
dhas2332	dahram777@hotmail.com	asdffa
jainish45	jainishah@gmail.com	jainishah
jas789	jashah@gmail.com	jas
kevin7	kevishah@gmail.com	kevin
kuldeep999	kuldeeepshah77@gmail.com	kuldeep
param342	paramshah999@hotmail.com	param
parth2	parthshah@gmail.com	parth
virti456	virtishah775@gmail.com	Virti

2)Display all the customers who did prepaid recharge with their plans and amount?

QUERY:

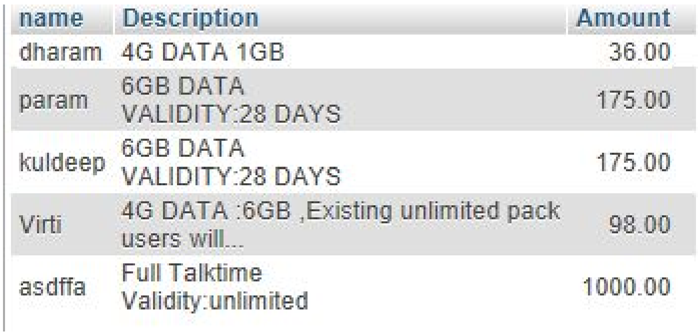
SELECT c.name, pa.Description,pa.Amount from customer c

JOINdoes d on d.username=c.username

JOIN prepaid p on p.rid=d.rid

JOIN plans pa on pa.pid=p.pid;

OUTPUT:



name	Description	Amount
dharam	4G DATA 1GB	36.00
param	6GB DATA VALIDITY:28 DAYS	175.00
kuldeep	6GB DATA VALIDITY:28 DAYS	175.00
Virti	4G DATA :6GB ,Existing unlimited pack users will...	98.00
asdffa	Full Talktime Validity:unlimited	1000.00

3)Find the customer who did bsnl recharge?

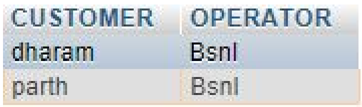
QUERY:

SELECT c.name CUSTOMER,o.name as OPERATOR from customer c

JOIN selects s on s.username=c.username

JOIN operator o on o.oid=s.oid where o.name='Bsnl';

OUTPUT:



CUSTOMER	OPERATOR
dharam	Bsnl
parth	Bsnl

4)Name all the admin who add plans for jio?

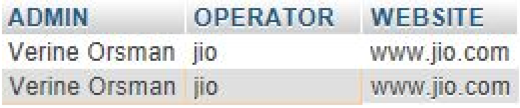
QUERY:

SELECT a.name as ADMIN,o.name OPERATOR ,o.site as WEBSITE from admin a

JOIN adds ad on ad.username=a.username

JOIN operator o on o.oid=ad.oid where o.name='jio'

OUTPUT:



ADMIN	OPERATOR	WEBSITE
Verine Orsman	jio	www.jio.com
Verine Orsman	jio	www.jio.com

5)Display all the 4g plans available for BSNL?

QUERY:

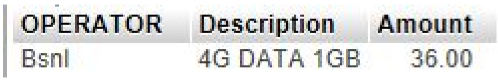
SELECT o.name OPERATOR,p.Description,p.Amount from data d

JOIN adds ad on d.pid=ad.pid

JOIN plans p on p.pid=d.pid

JOIN operator o on o.oid=ad.oid where o.name='Bsnl';

OUTPUT:



OPERATOR	Description	Amount
Bsnl	4G DATA 1GB	36.00

6)Display all the plans for Bsnl?

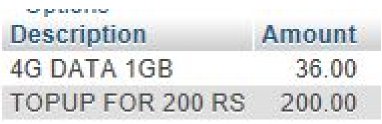
QUERY:

SELECT p.description as Description,p.amount as Amount from adds a

JOIN operator o on o.oid=a.oid

JOIN plans p on p.pid=a.pid where o.name='Bsnl';

OUTPUT:



Description	Amount
4G DATA 1GB	36.00
TOPUP FOR 200 RS	200.00

7)Display all the plans which were added by admin 'Clemente Longcake'?

QUERY:

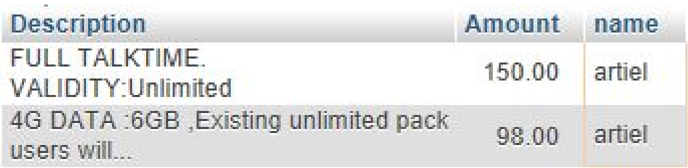
SELECT p.Description,p.Amount,o.name from adds a

JOIN admin ad on ad.username=a.username

JOIN plans p on p.pid=a.pid

JOIN operator o on o.oid=a.oid where ad.name='Clemente Longcake';

OUTPUT:



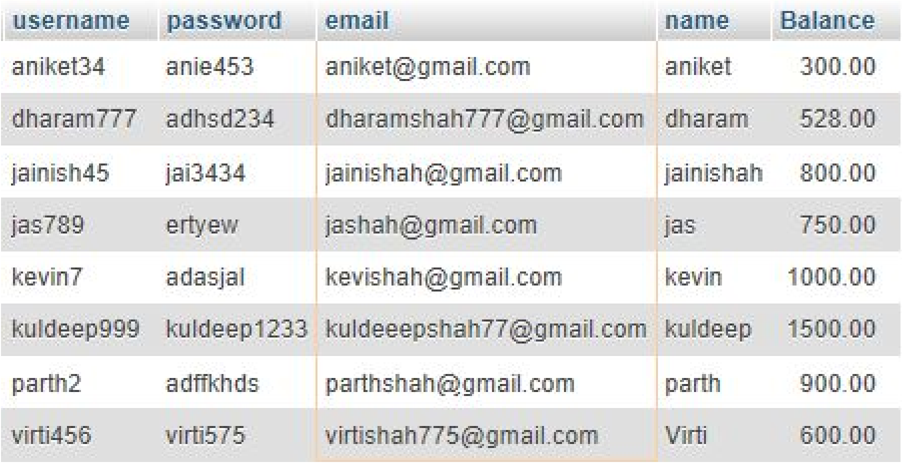
Description	Amount	name
FULL TALKTIME. VALIDITY:Unlimited	150.00	artiel
4G DATA :6GB ,Existing unlimited pack users will...	98.00	artiel

8)Name of all the customers whose account is on gmail?

QUERY:

SELECT * from customer where email LIKE"%gmail%"

OUTPUT:



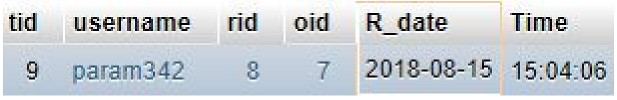
username	password	email	name	Balance
aniket34	anie453	aniket@gmail.com	aniket	300.00
dharam777	adhsd234	dharamshah777@gmail.com	dharam	528.00
jainish45	jai3434	jainishah@gmail.com	jainishah	800.00
jas789	ertyew	jashah@gmail.com	jas	750.00
kevin7	adasjal	kevishah@gmail.com	kevin	1000.00
kuldeep999	kuldeep1233	kuldeeepshah77@gmail.com	kuldeep	1500.00
parth2	adffkhds	parthshah@gmail.com	parth	900.00
virti456	virti575	virtishah775@gmail.com	Virti	600.00

9)All the customer who did recharge in August?

QUERY:

SELECT * from transaction_history where extract(MONTH from R_date)=8

OUTPUT:



tid	username	rid	oid	R_date	Time
9	param342	8	7	2018-08-15	15:04:06

10)Give name of all the customers whose recharge amount is greater than 300?

QUERY:

SELECT c.name as NAME ,c.username as USERNAME from transaction_history t

JOIN customer c on c.username=t.username

JOIN postpaid p on p.rid=t.rid where p.Amount>300

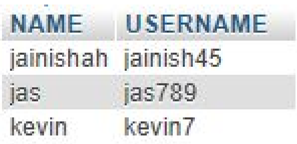
UNION select c.name as NAME ,c.username as USERNAME from transaction_history t

JOIN customer c on c.username=t.username

JOIN prepaid p on p.rid=t.rid

JOIN plans pa on pa.pid=p.pid where pa.Amount>300

OUTPUT:



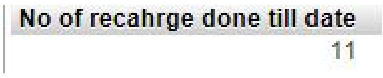
NAME	USERNAME
jainishah	jainish45
jas	jas789
kevin	kevin7

11) Total number of successful recharge till date?

QUERY:

```
SELECT COUNT(tid) as "No of recahrge done till date" from transaction_history
```

OUTPUT:



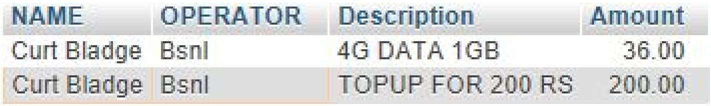
No of recahrge done till date
11

14)Name the admin who added plans for Bsnl add what type of plan was added by him?

QUERY:

```
SELECT a.name,o.name,pl.Description,pl.Amount from admin a
JOIN adds ad on a.username=ad.username
JOIN operator o on o.oid=ad.oid
JOIN plans pl on pl.pid=ad.pid where o.name='Bsnl'
```

OUTPUT:



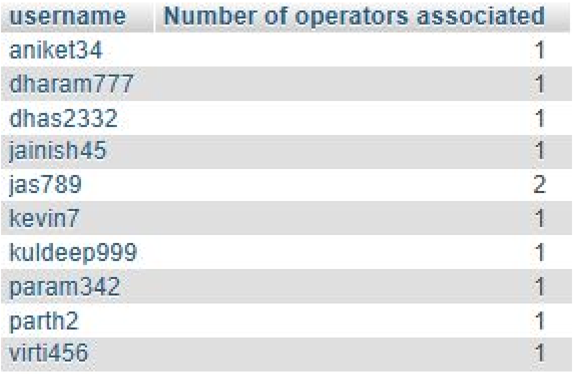
NAME	OPERATOR	Description	Amount
Curt Bladge	Bsnl	4G DATA 1GB	36.00
Curt Bladge	Bsnl	TOPUP FOR 200 RS	200.00

15)How many operators are associated with each customer?

QUERY:

```
SELECT username,COUNT(oid)
as 'Number of operators associated'
from selects GROUP by (username)
```

OUTPUT:



username	Number of operators associated
aniket34	1
dharam777	1
dhas2332	1
jainish45	1
jas789	2
kevin7	1
kuldeep999	1
param342	1
parth2	1
virti456	1

16)GIve number of customer whose recharge done before 10'o clock.

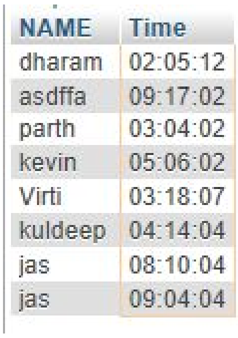
QUERY:

SELECT c.name as NAME,t.Time from customer c

JOIN transaction_history t on t.username=c.username

where extract(hour from t.Time)<10

OUTPUT:



NAME	Time
dharam	02:05:12
asdffa	09:17:02
parth	03:04:02
kevin	05:06:02
Virti	03:18:07
kuldeep	04:14:04
jas	08:10:04
jas	09:04:04

17)Give all the information regarding recharge and customer on operator.

QUERY:

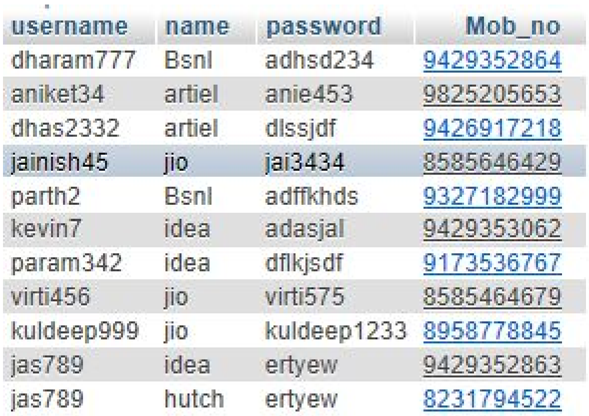
SELECT c.username,o.name,c.password,r.Mob_no from customer c

JOIN transaction_history t on t.username=c.username

JOIN recharge r on r.rid=t.rid

JOIN operator o on o.oid=t.oid;

OUTPUT:



username	name	password	Mob_no
dharam777	Bsnl	adhsd234	9429352864
aniket34	artiel	anie453	9825205653
dhas2332	artiel	dlssjdf	9426917218
jainish45	jio	jai3434	8585646429
parth2	Bsnl	adffkhds	9327182999
kevin7	idea	adasjal	9429353062
param342	idea	dflkjsdf	9173536767
virti456	jio	virti575	8585464679
kuldeep999	jio	kuldeep1233	8958778845
jas789	idea	ertyew	9429352863
jas789	hutch	ertyew	8231794522

18)Display prepaid customer name and recharge amount who has done highest amount recharge.

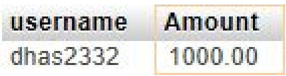
QUERY:

SELECT d.username,pa.Amount from does d

JOIN recharge r on r.rid=d.rid

```
JOIN prepaid pr on pr.rid=r.rid
JOIN plans pa on pa.pid=pr.pid
where pa.pid=any(
SELECT pa.pid from plans pa where pa.Amount=(
SELECT MAX(pa.Amount) from plans pa))
```

OUTPUT:



username	Amount
dhas2332	1000.00

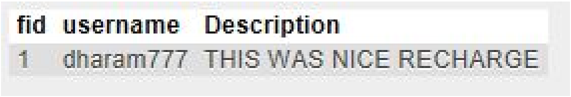
5.4 PL/SQL BLOCKS

1)Procedure for getting feedback given by specific customer by providing username.

PL/SQL Program:

```
DELIMITER $$
CREATE PROCEDURE `getFEEDBACK`(IN given_username varchar(25))
BEGIN
SELECT *  FROM feedback where username=given_username ;
END$$
DELIMITER ;
```

OUTPUT:



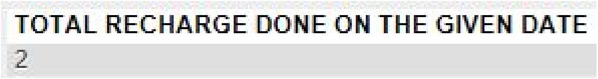
fid	username	Description
1	dharam777	THIS WAS NICE RECHARGE

2) Procedure for total number of recharge on the given date.

PL/SQL Program:

```
DELIMITER $$
CREATE PROCEDURE `total_RECHARGE`(IN `dt` DATE)
BEGIN
DECLARE VAR1 INT(100) DEFAULT 0;
SELECT COUNT(tid)  INTO VAR1 FROM transaction_history where R_date=dt;
SELECT VAR1 AS 'TOTAL RECHARGE DONE ON THE GIVEN DATE' ;
END$$
DELIMITER ;
```

OUTPUT:



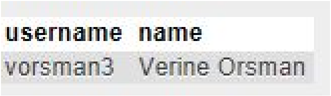
TOTAL RECHARGE DONE ON THE GIVEN DATE
2

3)Procedure for name of admin who added plan for specific operator.

PL/SQL Program:

```
DELIMITER $$
CREATE  PROCEDURE `Admin_for_operator`(IN`operator_name`VARCHAR(15))
BEGIN
SELECT DISTINCT a.username,a.name  FROM admin a
JOIN adds ad on ad.username=a.username
JOIN operator o  on o.oid=ad.oid where o.name=operator_name;
END$$
DELIMITER ;
```

OUTPUT:



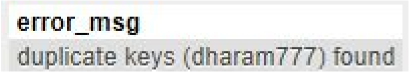
username	name
vorsman3	Verine Orsman

4) Procedure for insert into customer raises exception if it finds same username.

PL/SQL Program:

```
DELIMITER $$
CREATE  PROCEDURE `insert_CUSTOMER`
(IN username varchar(15),IN password varchar(15), IN email varchar(25), IN name
char(20))
BEGIN
DECLARE CONTINUE HANDLER FOR 1062
SELECT CONCAT('duplicate keys (',username,') found') AS error_msg;
INSERT INTO customer values(username,password,email,name);
END$$
DELIMITER ;
```

OUTPUT:



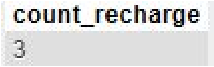
```
error_msg
duplicate keys (dharam777) found
```

5)Function which calculates total recharges done by the given operator.

PL/SQL Program:

```
DELIMITER $$
CREATE  FUNCTION `count_recharge`(`oname` VARCHAR(15)) RETURNS int(10)
   DETERMINISTIC
BEGIN
 DECLARE TOTAL INT(10);
SELECT COUNT(s.oid) as TOTAL_RECHARGE INTO TOTAL from selects s
JOIN operator o on o.oid=s.oid WHERE o.name=oname;
RETURN(TOTAL);
END$$
DELIMITER ;
```

OUTPUT:



```
count_recharge
3
```

6)Procedure to find that the given user did prepaid or postpaid recharge.

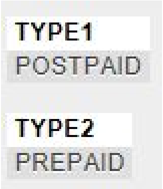
PL/SQL Program:

```
DELIMITER $$
CREATE PROCEDURE `POST_PRE`(IN `username` VARCHAR(11))
BEGIN
DECLARE VAR1 INT (10);
DECLARE VAR2 INT (10);
DECLARE TYPE1 VARCHAR(10);
DECLARE TYPE2 VARCHAR(10);
 SELECT r.rid into VAR1 from does d
JOIN postpaid r on r.rid=d.rid where d.username=username;
 SELECT r.rid into VAR2 from does d
JOIN prepaid r on r.rid=d.rid where d.username=username;

IF VAR1!=0 THEN
```

```
SET TYPE1="POSTPAID";
END IF;
IF VAR2!=0 THEN
SET TYPE2="PREPAID";
 END IF;
SELECT TYPE1;
SELECT TYPE2;
END$$
DELIMITER ;
```

OUTPUT:



```
TYPE1
POSTPAID

TYPE2
PREPAID
```

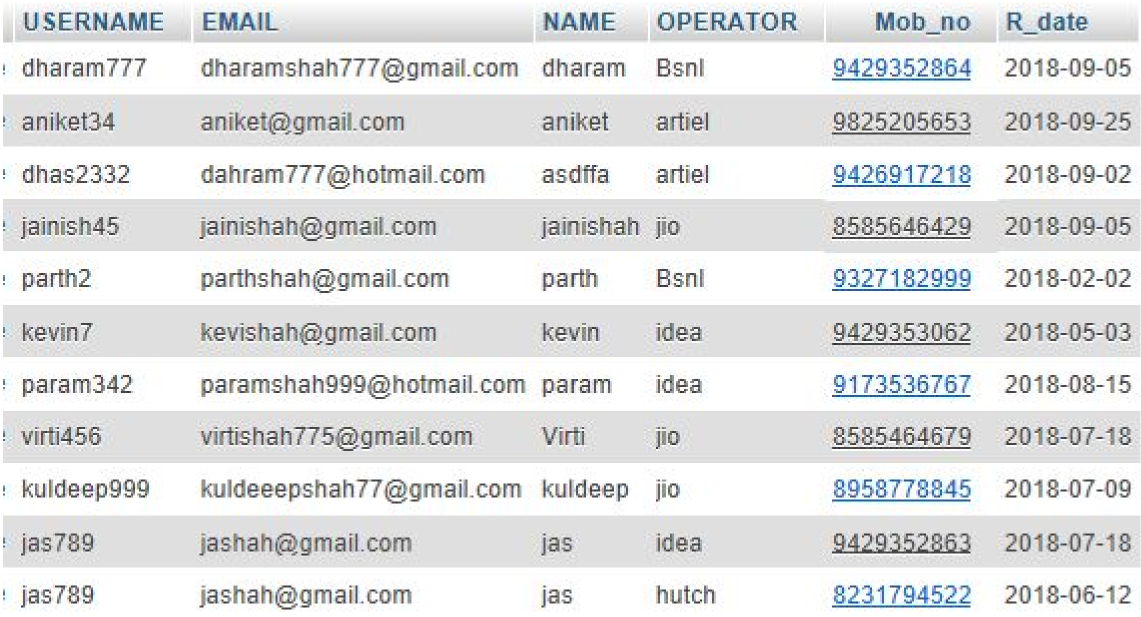
5.5 VIEWS

1) View on customer's transaction history with date and time.

CREATE QUERY:

```
CREATE view customer_transaction as
SELECT c.username as USERNAME,c.email as EMAIL,c.name as NAME,o.name as
OPERATOR,r.Mob_no,t.R_date from customer c
 JOIN transaction_history t on t.username=c.username
JOIN operator o on o.oid=t.oid
JOIN  recharge r on r.rid=t.rid
```

OUTPUT:



USERNAME	EMAIL	NAME	OPERATOR	Mob_no	R_date
dharam777	dharamshah777@gmail.com	dharam	Bsnl	9429352864	2018-09-05
aniket34	aniket@gmail.com	aniket	artiel	9825205653	2018-09-25
dhas2332	dahram777@hotmail.com	asdffa	artiel	9426917218	2018-09-02
jainish45	jainishah@gmail.com	jainishah	jio	8585646429	2018-09-05
parth2	parthshah@gmail.com	parth	Bsnl	9327182999	2018-02-02
kevin7	kevishah@gmail.com	kevin	idea	9429353062	2018-05-03
param342	paramshah999@hotmail.com	param	idea	9173536767	2018-08-15
virti456	virtishah775@gmail.com	Virti	jio	8585464679	2018-07-18
kuldeep999	kuldeeepshah77@gmail.com	kuldeep	jio	8958778845	2018-07-09
jas789	jashah@gmail.com	jas	idea	9429352863	2018-07-18
jas789	jashah@gmail.com	jas	hutch	8231794522	2018-06-12

2)View for which of the operator plan was added by which customer?

CREATE QUERY:

CREATE view ADMIN_OPERATOR as

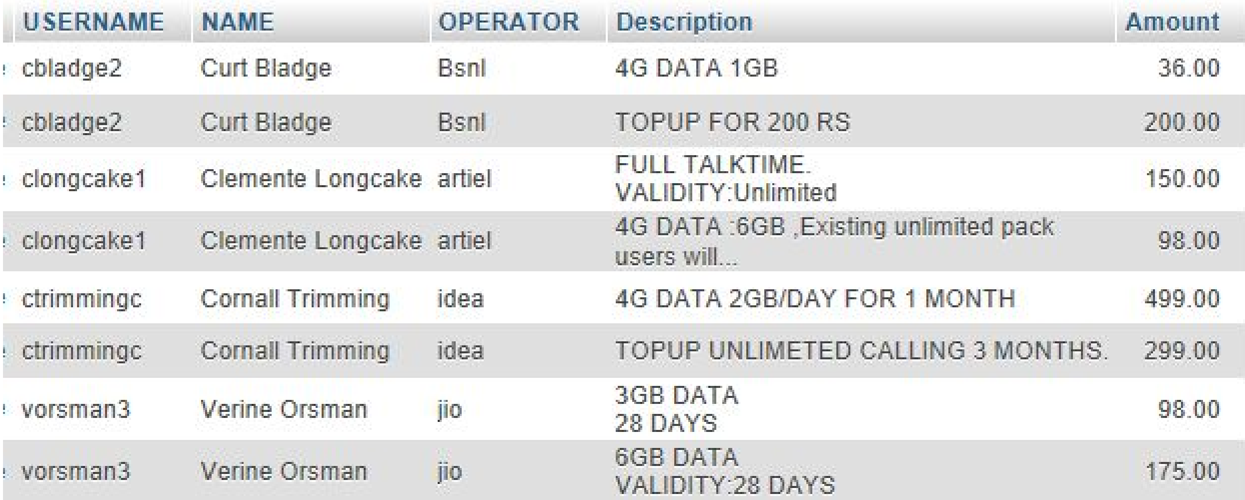
SELECT a.username as USERNAME, a.name as NAME , o.name as OPERATOR,p.Description,p.Amount from admin a

JOIN adds ad on ad.username=a.username

JOIN operator o on ad.oid=o.oid

JOIN plans p on p.pid=ad.pid

OUTPUT:



USERNAME	NAME	OPERATOR	Description	Amount
cbladge2	Curt Bladge	Bsnl	4G DATA 1GB	36.00
cbladge2	Curt Bladge	Bsnl	TOPUP FOR 200 RS	200.00
clongcake1	Clemente Longcake	artiel	FULL TALKTIME. VALIDITY:Unlimited	150.00
clongcake1	Clemente Longcake	artiel	4G DATA :6GB ,Existing unlimited pack users will...	98.00
ctrimmingc	Cornall Trimming	idea	4G DATA 2GB/DAY FOR 1 MONTH	499.00
ctrimmingc	Cornall Trimming	idea	TOPUP UNLIMETED CALLING 3 MONTHS.	299.00
vorsman3	Verine Orsman	jio	3GB DATA 28 DAYS	98.00
vorsman3	Verine Orsman	jio	6GB DATA VALIDITY:28 DAYS	175.00

5.6 TRIGGERS

1)Deduct the balance of postpaid customer from as soon as he does the recharge.If the customer does not have enough balance the generate error.

PL/SQL Program:

```
CREATE TRIGGER `before_postpaid_insert` BEFORE INSERT ON `postpaid`
 FOR EACH ROW BEGIN
DECLARE var2 int(10);

SELECT Balance into var2 from customer c where c.username=any(
SELECT d.username from does d
JOIN postpaid p on NEW.rid=d.rid  );
IF NEW.Amount<=var2 then
update customer c set c.Balance=c.Balance-NEW.Amount
where c.username=any(
SELECT d username from does d
JOIN postpaid p on NEW.rid=d.rid );
 ELSE
 CALL`"INSUFFICIENT BALANCE"`;
 END IF;
END
```

2)Deduct the balance of prepaid customer after he selects the plan.

PL/SQL Program:

```
CREATE TRIGGER `before_prepaid_insert` AFTER INSERT ON `prepaid`
 FOR EACH ROW BEGIN
DECLARE VAR1 int(6);
DECLARE var2 decimal(6,2);
SELECT Balance into VAR1 from customer c where c.username=any(
SELECT d.username from does d
JOIN prepaid p on NEW.rid=d.rid where p.pid=NEW.pid );
SELECT Amount into var2 from plans p
```

```
JOIN prepaid pr on NEW.pid=p.pid where pr.rid=NEW.rid;

IF var2<=VAR1  THEN
          UPDATE customer c SET c.Balance=c.Balance-var2
          where c.username=any(
SELECT d.username from does d
JOIN prepaid p on NEW.rid=d.rid where p.pid=NEW.pid);
          ELSE
   call`"insuffcient balance"`;
END IF;
END
```

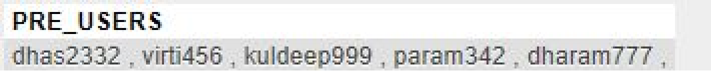
5.7 CURSORS

1)Display all prepaid users who has done recharge.

PL/SQL Program:

```
DELIMITER $$
CREATE  PROCEDURE `PRE_USERS`(INOUT `PRE_USERS` VARCHAR(300))
BEGIN
DECLARE VAR2 varchar(20) DEFAULT "";
DECLARE FINISHED2 INTEGER DEFAULT 0;
DECLARE CURSOR2 CURSOR FOR
SELECT d.username  from does d join prepaid r on r.rid=d.rid ;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET FINISHED2=TRUE ;
OPEN CURSOR2;
LOOP2:LOOP
FETCH CURSOR2 INTO VAR2;
IF FINISHED2 THEN
LEAVE LOOP2;
END IF;
SET PRE_USERS=CONCAT(VAR2," , ",PRE_USERS);
END LOOP LOOP2;
CLOSE CURSOR2;
END$$
```

```
DELIMITER ;
```

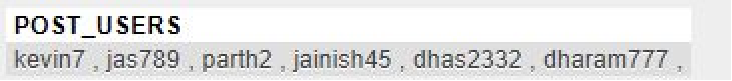


```
PRE_USERS
dhas2332 , virti456 , kuldeep999 , param342 , dharam777 ,
```

2)Display all postpaid users who has done recharge.

PL/SQL Program:

```
DELIMITER $$
CREATE POST_USERS`(INOUT `POST_USERS` VARCHAR(300))
BEGIN
DECLARE VAR1 varchar(20) DEFAULT "";
DECLARE FINISHED1 INTEGER DEFAULT 0;
DECLARE  CURSOR1 CURSOR FOR
SELECT d.username  from does d join postpaid r on r.rid=d.rid;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET FINISHED1=TRUE;
OPEN CURSOR1;
LOOP1:LOOP
FETCH CURSOR1 INTO VAR1;
IF FINISHED1 THEN
LEAVE LOOP1;
END IF;
SET POST_USERS=CONCAT(VAR1," , ",POST_USERS);
END LOOP LOOP1;
CLOSE CURSOR1;
END$$
DELIMITER ;.
```



```
POST_USERS
kevin7 , jas789 , parth2 , jainish45 , dhas2332 , dharam777 ,
```

FUTURE ENHANCEMENTS OF THE SYSTEM

This system will be very useful as the world is enhancing towards the era of digital system so as to recharge from anywhere online in just a second. Helps customer to have immediate recharge information like tarrif without going anywhere.

BIBILIOGRAPHY

- Database System Concepts - Fourth Edition by Silberschatz-Korth-Sudarshan

- Fundamentals of Database System –Shamkant B. Navathe