American International University-Bangladesh (AIUB)

AMERICAN INTERNATIONAL UNIVERSITY-BANGLADESH (AIUB)

Undergraduate Program

Course: ADVANCE DATABASE MANAGEMENT SYSTEM **Summer 2021**

Title: Prison Management System

Submitted by :

Name	ID
Oishi Chowdhury	**18-36807-1**

Submitted To

JUENA AHMED NOSHIN

Lecturer
Faculty of Science and Technology
American International University-Bangladesh

Contents

1. Introduction:

Management of prisons in Nigeria has long been a neglected area which has recently been incorporated in the e-governance program of Government of Nigeria. Currently a rudimentary process of storing all the prisoner data in manual files and registers is in place. The Prison Management System project will integrate all the prisoner data into a single integrated system which will in turn result all the information being present in a digital format.

ICT in prisons was initiated in the year 2002 at Delhi Prisons, Tihar. The Tihar Prisons Complex in New Delhi is the biggest prison complex in Asia comprising of 9 prisons and one District Jail at Rohini with a total strength of more than 11,000 prisoners against a normal sanctioned capacity of 6250 prisoners. In a year about 70,000 – 80,000 inmates remain lodged in these prisons for different duration and crimes committed by them. This prison population has about 80% under trials and includes about 480 women inmates. About 400 inmates are foreigners from different parts of the world. Many high security criminals also live here. There has been a substantial increase in number of prison inmates coming to Tihar because of a phenomenal increase in the crime scene at Delhi that has resulted in the increase of the ICT needs and its management at the Tihar Jail Complex.

Nearly 1700-1800 visitors meet their relative inmates' everyday. There was manual system of booking (meetings) in each jail for its respective inmates. Centralised visitor record was not available. There was lack of exchange of visitors' information within jails and prison headquarters. No provision for identification / detection / verification of visitors was there.

Managing the prisoner record and monitoring of prisoner / visitor was always difficult since most of the records were normally maintained manually, so the concerned authorities were required to go through all the registers to find out the details and status of the inmate as well as of the visitor.

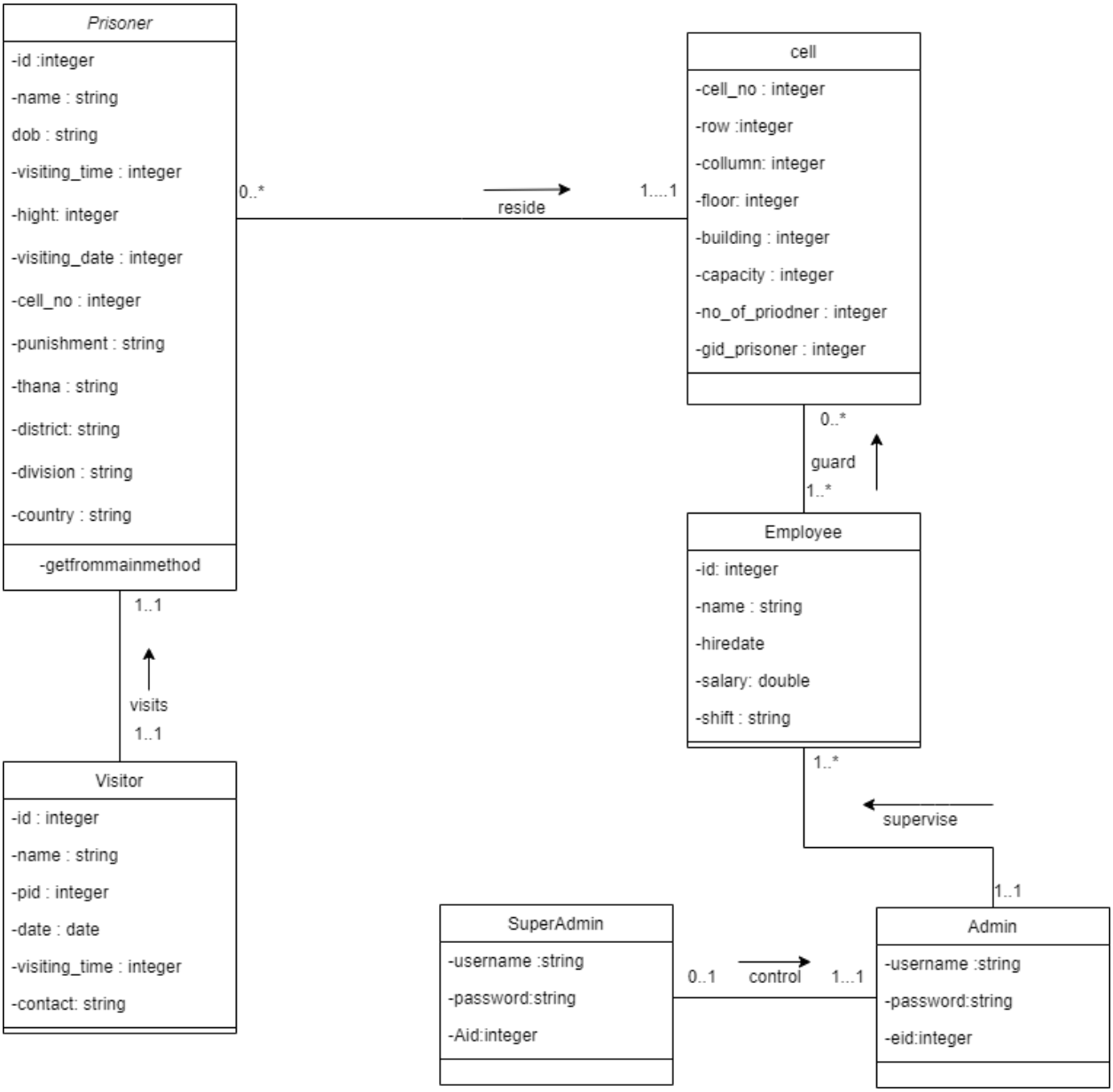
2.Project Proposal:

This application will help to store, update, monitor and manage all data of the entities of a prison. Utilizing that data through this system, the person in charge will be able to manage and run the prison easily and effectively. That's why it is a PRISON MANAGEMENT SYSTEM.

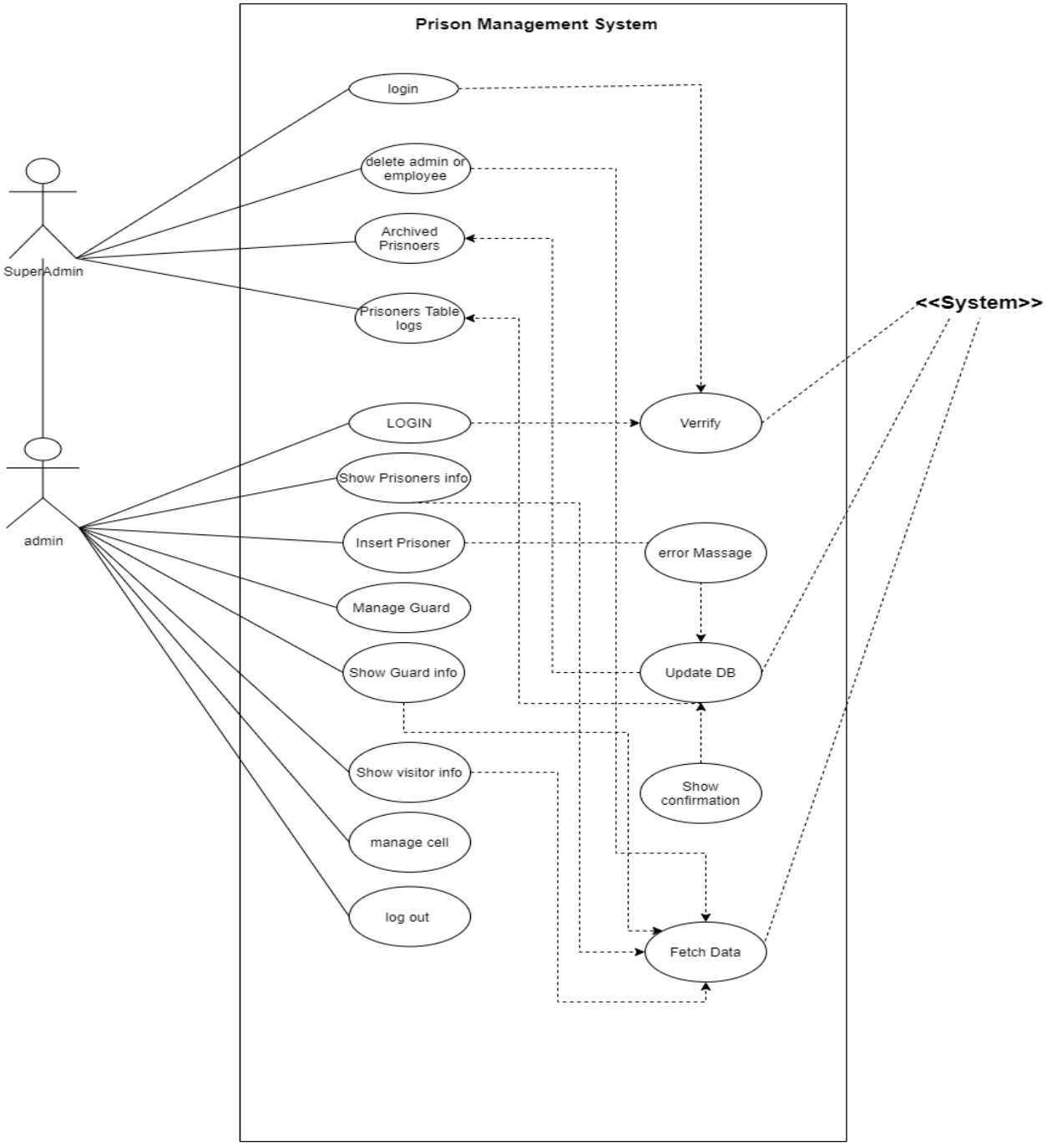
Firstly, to user the system the user is needed to be authenticated. After successful log in the user get to choose whether to insert or show information of a prisoner or to manage them. There are also options to manage cell, show employee information and manage employee. By choosing mange cell the user can see all cell related information and can assign-reassign prisoners to cells basing on availability and need. Through manage employee module the user can manage employee working shift and assign guard to cells accordingly. Through the system the authenticated person also can access to information of visitors who visited the prisoners so that that information can be utilized if they are ever needed in the future. Then if all the work have been finished, the user can log out from the system to prevent any un-authorized uses.

3.Diagram :

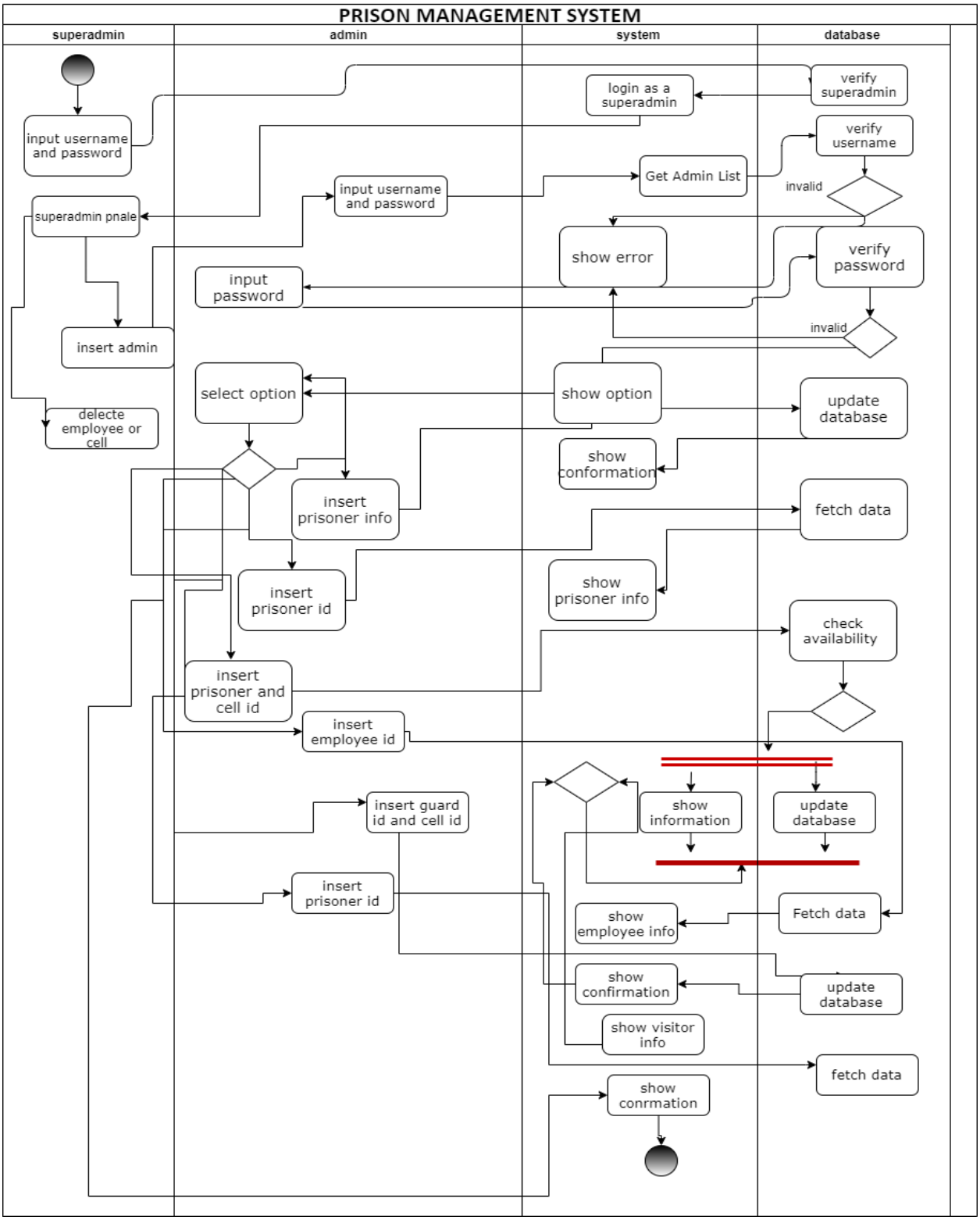
3.1.Class Diagram:



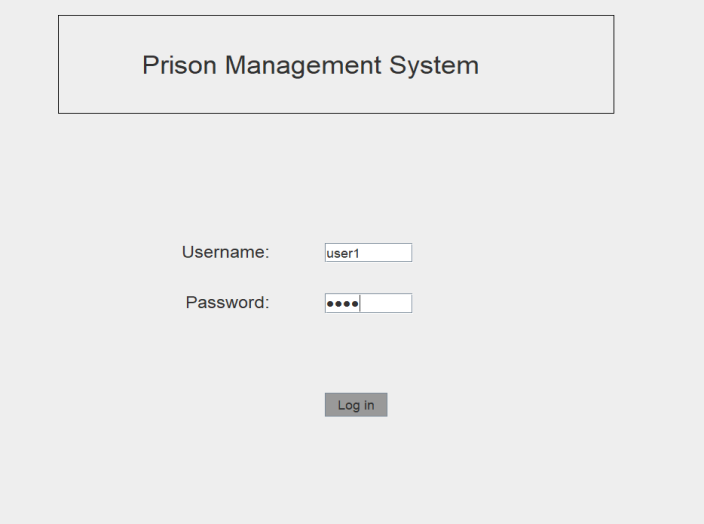
3.2. Use Case Diagram:

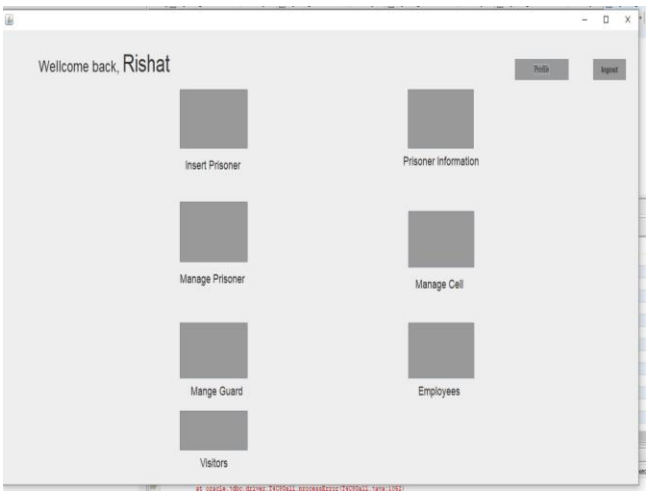


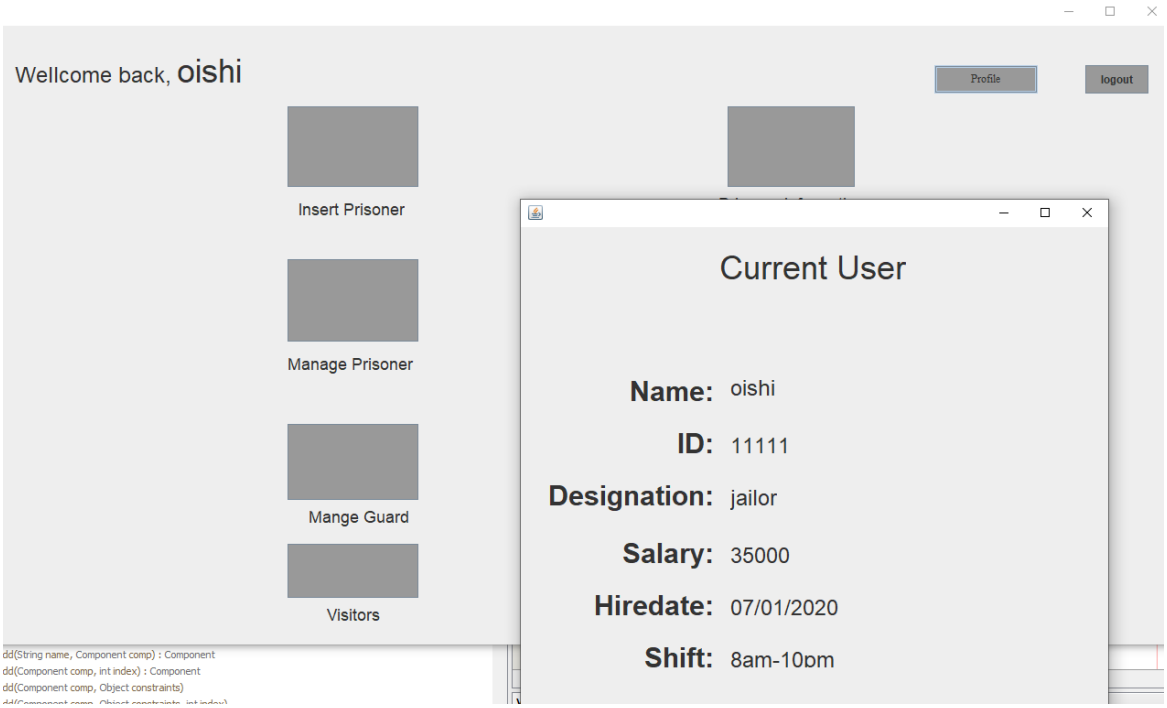
3.3. Activity Diagram

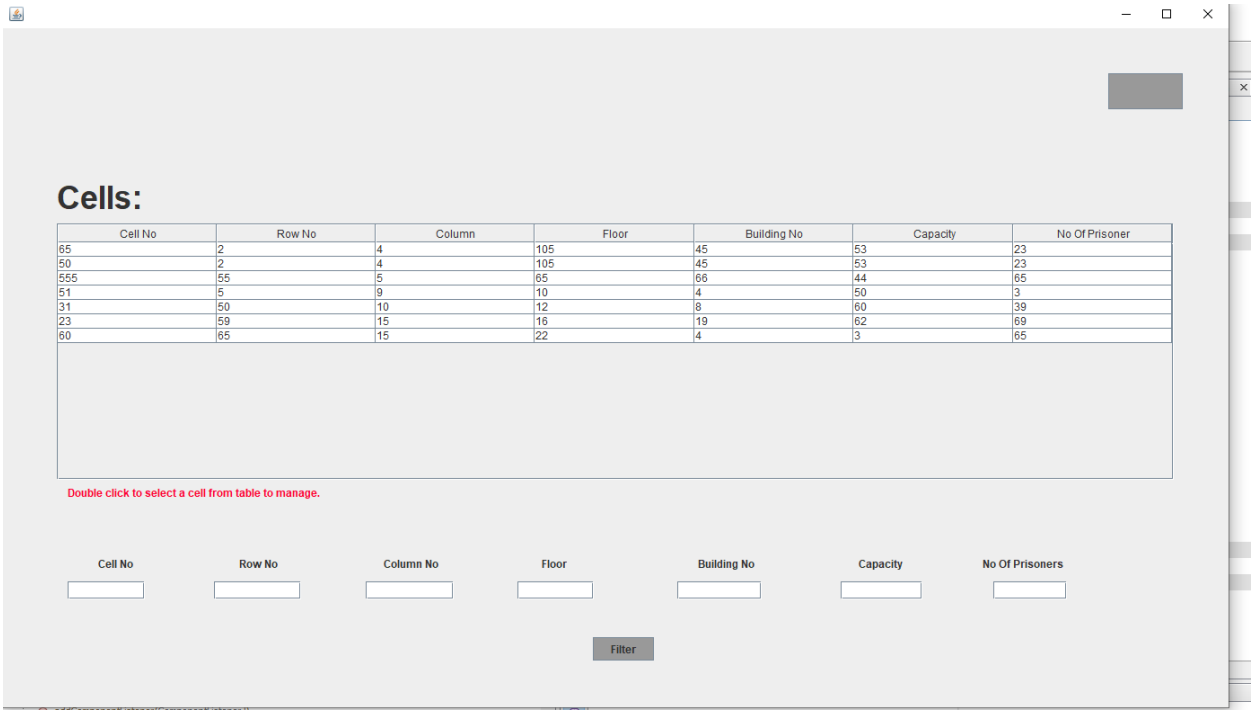


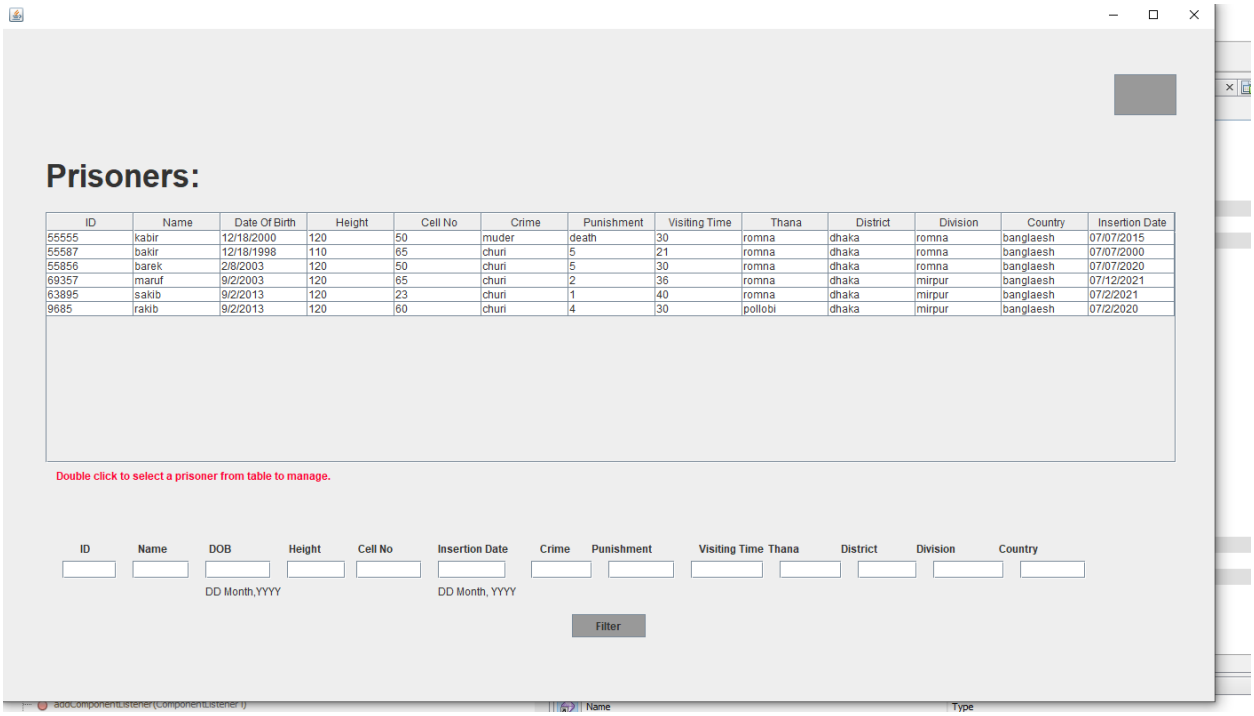
4. User Interface:

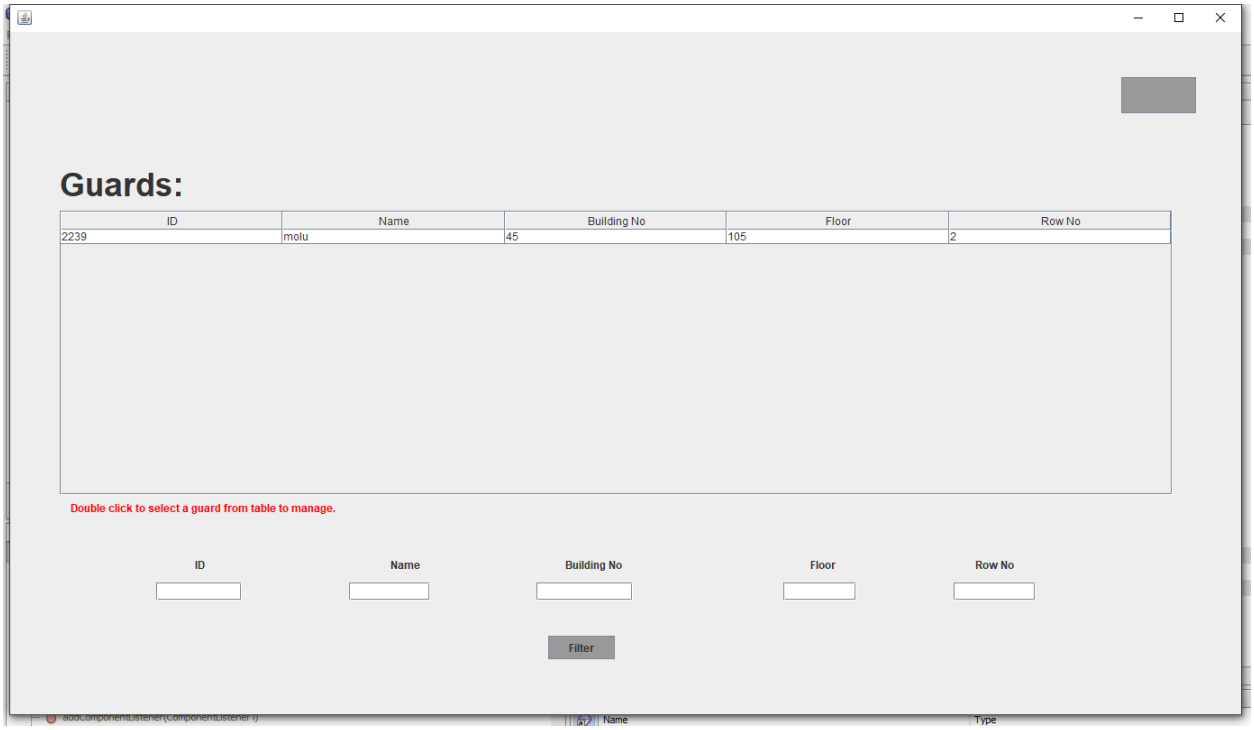


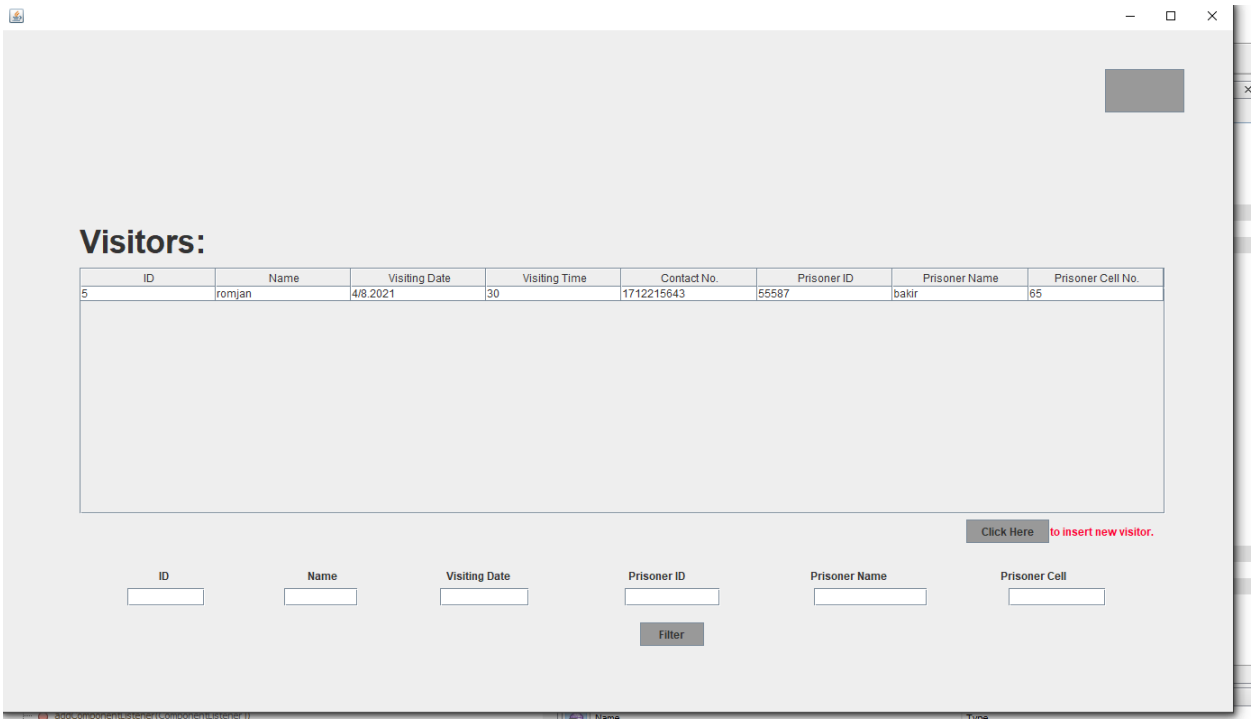


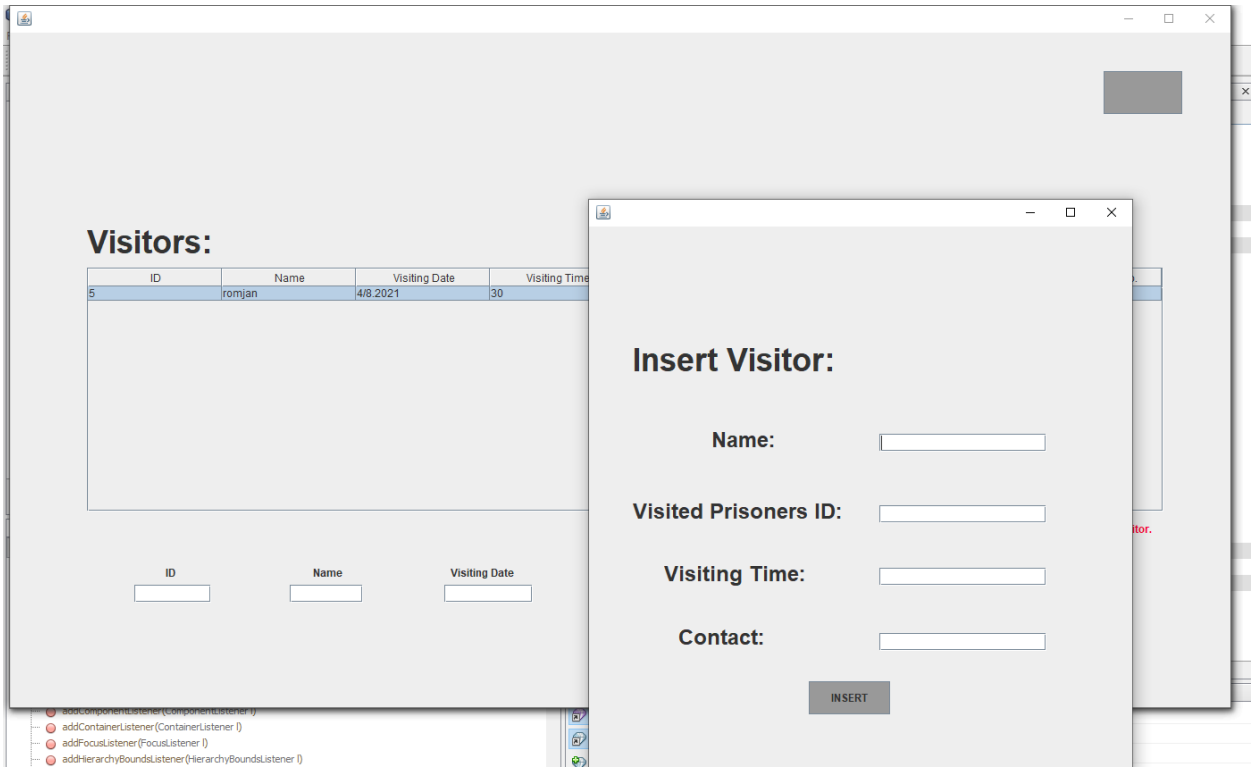


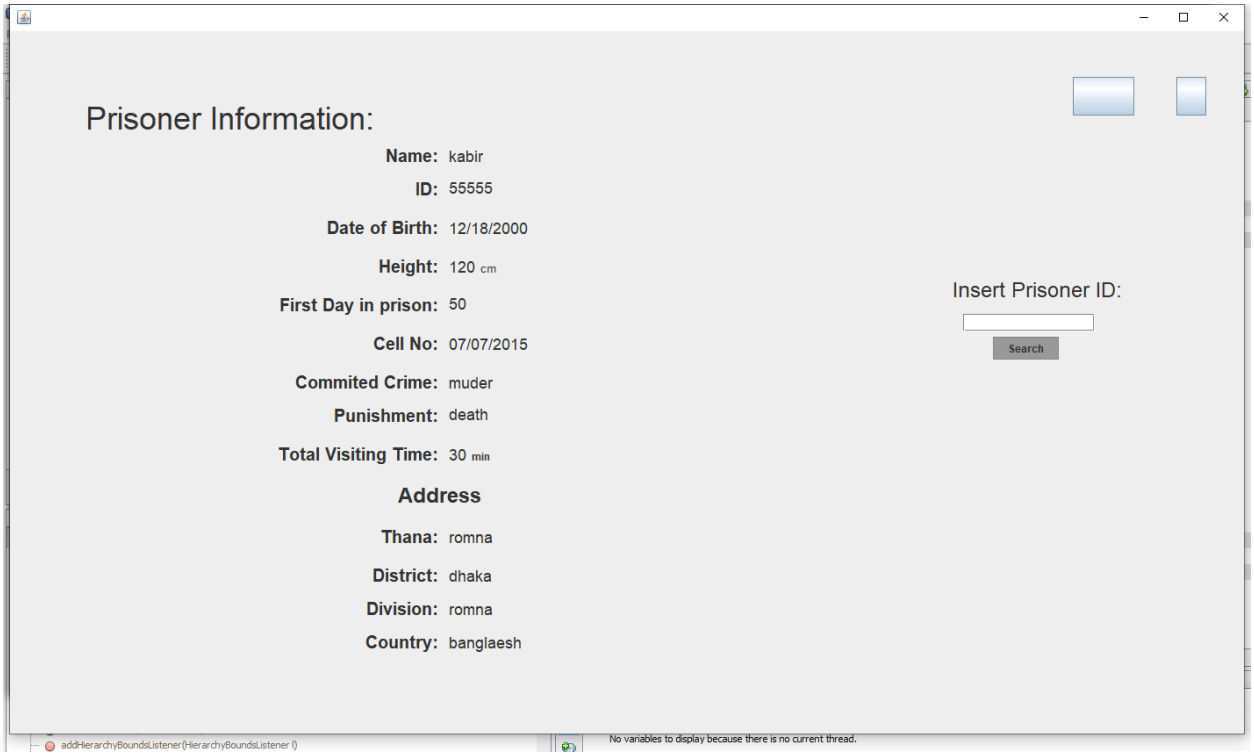

Cells:
Cell No | Row No | Column | Floor | Building No | Capacity | No Of Prisoner
65 | 2 | 4 | 105 | 45 | 53 | 23
50 | 2 | 4 | 105 | 45 | 53 | 23
555 | 55 | 5 | 65 | 66 | 44 | 65
51 | 5 | 9 | 10 | 4 | 50 | 3
31 | 50 | 10 | 12 | 8 | 60 | 39
23 | 59 | 15 | 16 | 19 | 62 | 69
60 | 65 | 15 | 22 | 4 | 3 | 65
Double click to select a cell from table to manage.
Cell No
Row No
Column No
Floor
Building No
Capacity
No Of Prisoners
Filter

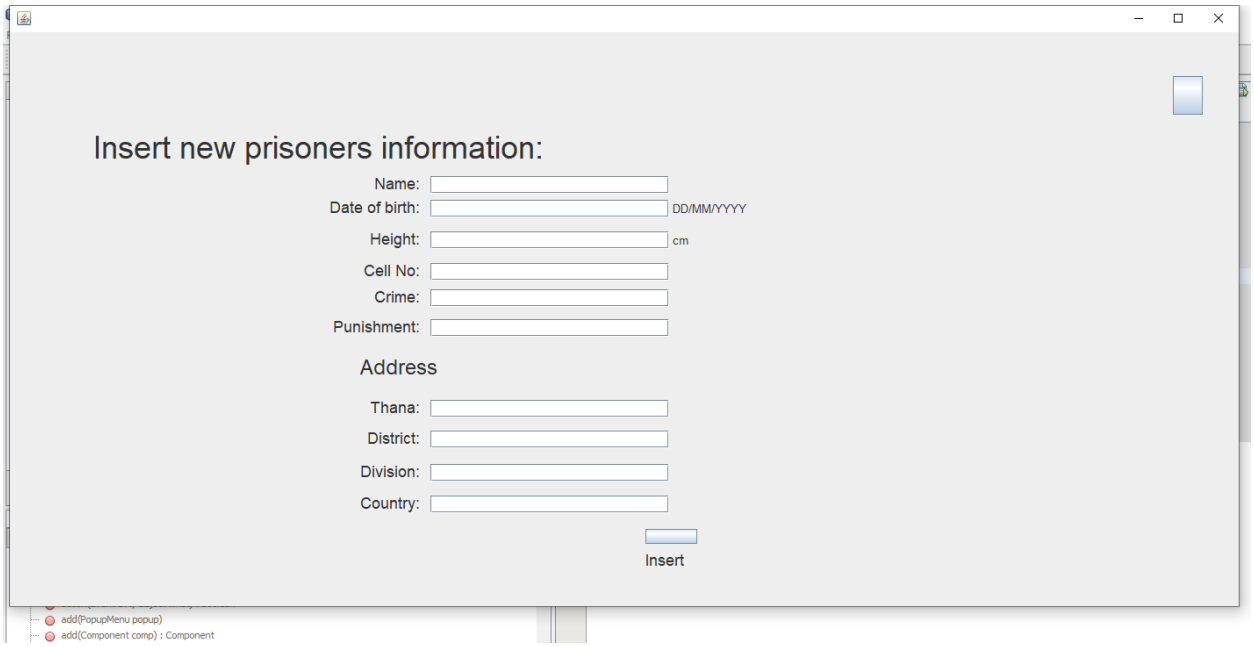

Prisoners:
ID | Name | Date Of Birth | Height | Cell No | Crime | Punishment | Visiting Time | Thana | District | Division | Country | Insertion Date
55555 | kabir | 12/18/2000 | 120 | 50 | muder | death | 30 | romna | dhaka | romna | banglaesh | 07/07/2015
55587 | bakir | 12/18/1998 | 110 | 65 | churi | 5 | 21 | romna | dhaka | romna | banglaesh | 07/07/2000
55856 | barek | 2/8/2003 | 120 | 50 | churi | 5 | 30 | romna | dhaka | romna | banglaesh | 07/07/2020
69357 | maruf | 9/2/2003 | 120 | 65 | churi | 2 | 36 | romna | dhaka | mirpur | banglaesh | 07/12/2021
63895 | sakib | 9/2/2013 | 120 | 23 | churi | 1 | 40 | romna | dhaka | mirpur | banglaesh | 07/2/2021
9685 | rakib | 9/2/2013 | 120 | 60 | churi | 4 | 30 | pollobi | dhaka | mirpur | banglaesh | 07/2/2020
Double click to select a prisoner from table to manage.
ID
Name
DOB
Height
Cell No
Insertion Date
Crime
Punishment
Visiting Time
Thana
District
Division
Country
DD Month,YYYY
DD Month, YYYY
Filter

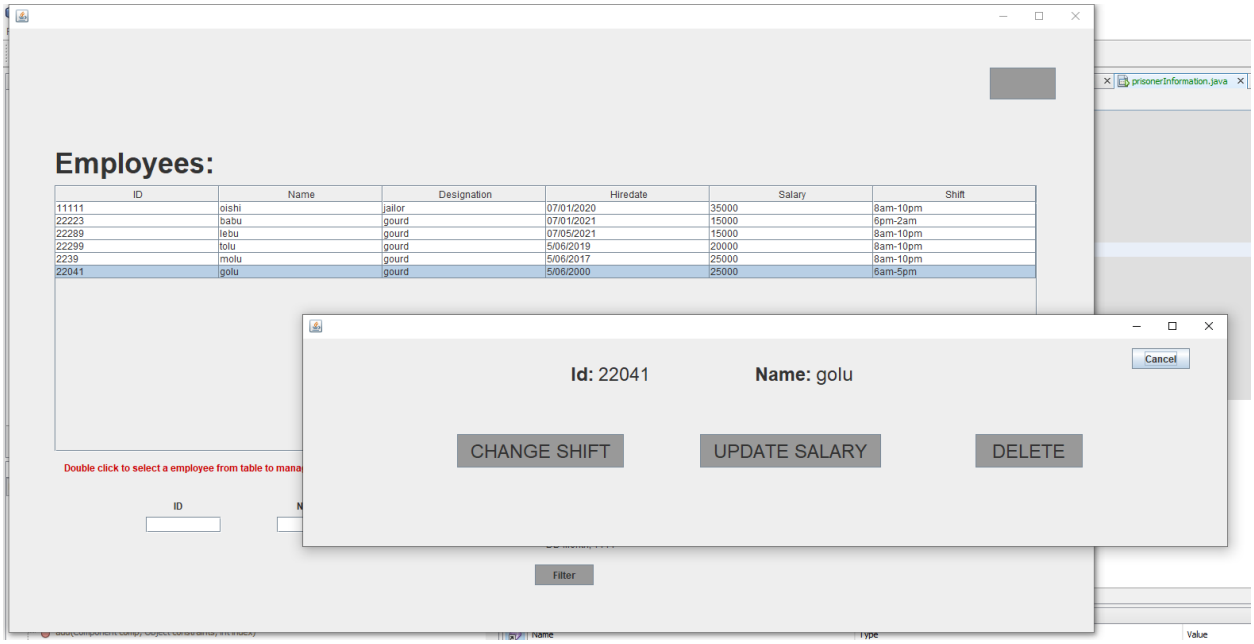

Guards:
ID
Name
Building No
Floor
Row No
2239
molu
45
105
2
Double click to select a guard from table to manage.
ID
Name
Building No
Floor
Row No
Filter

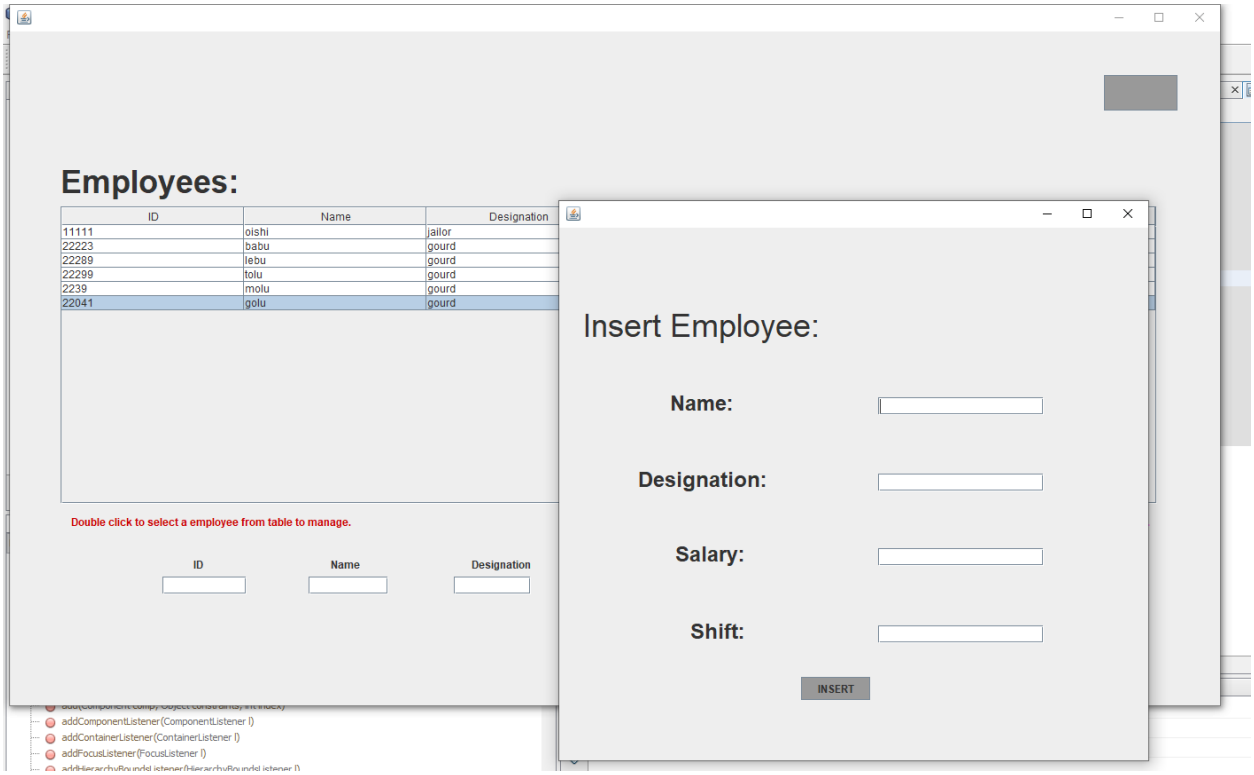

Visitors:
ID
Name
Visiting Date
Visiting Time
Contact No.
Prisoner ID
Prisoner Name
Prisoner Cell No.
5
romjan
4/8.2021
30
1712215643
55587
bakir
65
Click Here
to insert new visitor.
ID
Name
Visiting Date
Prisoner ID
Prisoner Name
Prisoner Cell
Filter

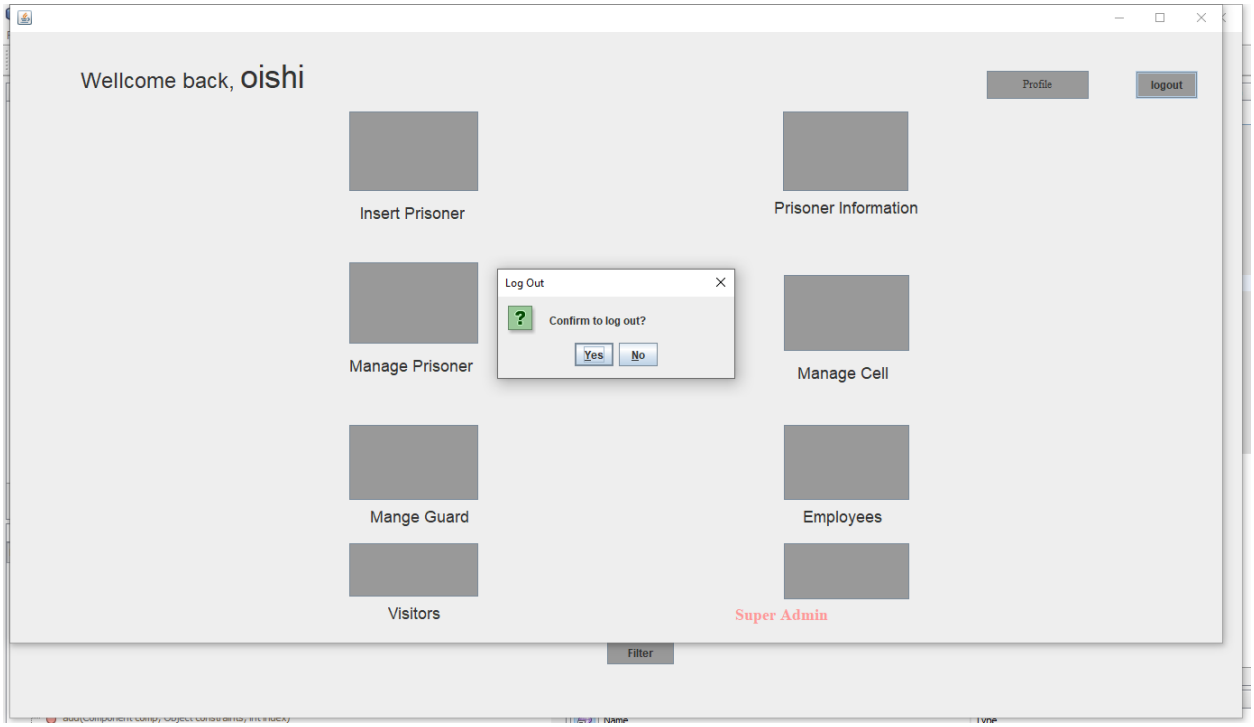

Visitors:
ID
Name
Visiting Date
Visiting Time
5
romjan
4/8.2021
30
ID
Name
Visiting Date
Insert Visitor:
Name:
Visited Prisoners ID:
Visiting Time:
Contact:
INSERT

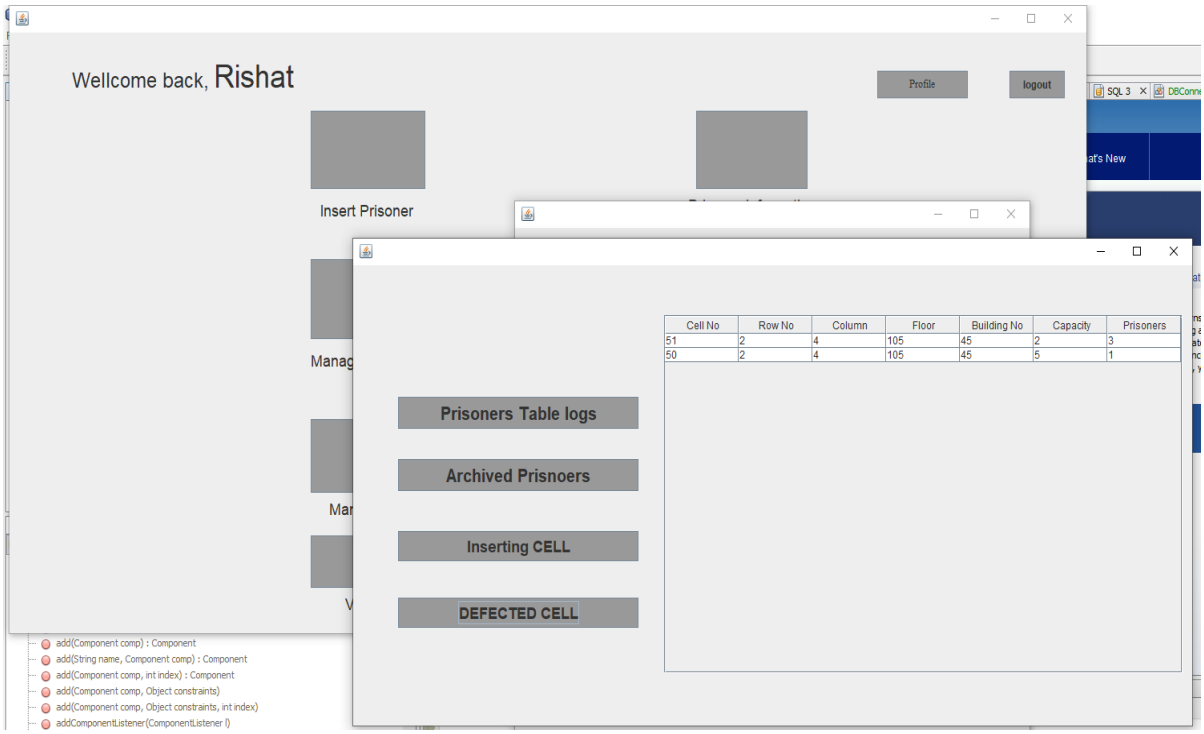

Prisoner Information:
Name: kabir
ID: 55555
Date of Birth: 12/18/2000
Height: 120 cm
First Day in prison: 50
Cell No: 07/07/2015
Commited Crime: muder
Punishment: death
Total Visiting Time: 30 min
Address
Thana: romna
District: dhaka
Division: romna
Country: banglaesh
Insert Prisoner ID:
Search

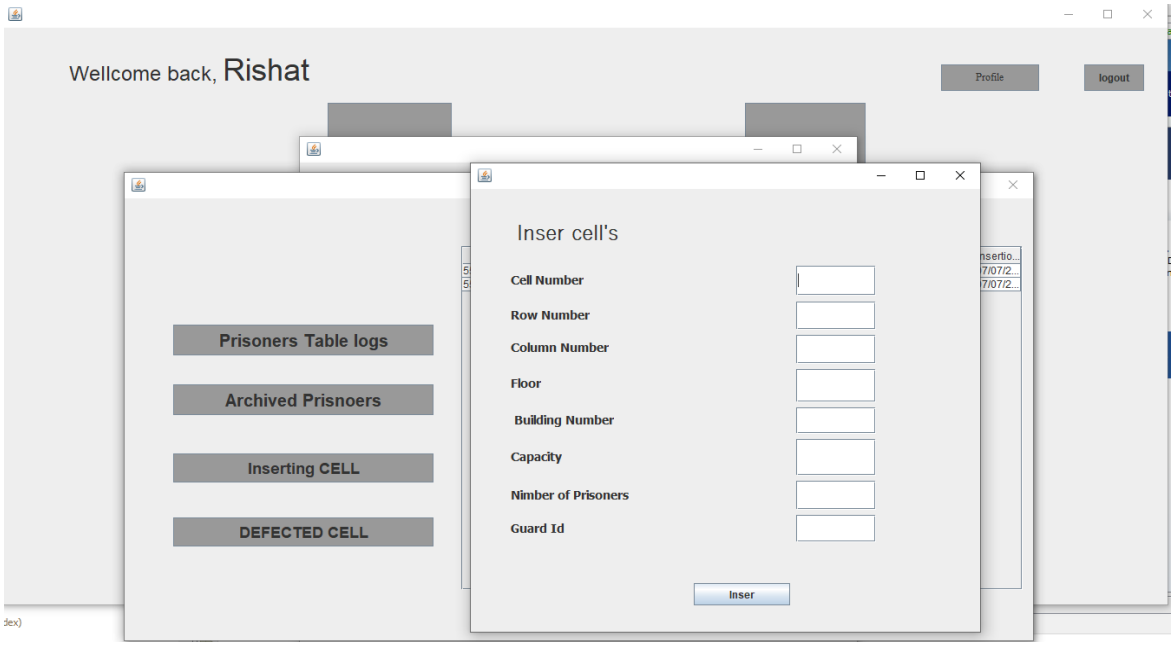

Insert new prisoners information:
Name:
Date of birth:
DD/MM/YYYY
Height:
cm
Cell No:
Crime:
Punishment:
Address
Thana:
District:
Division:
Country:
Insert

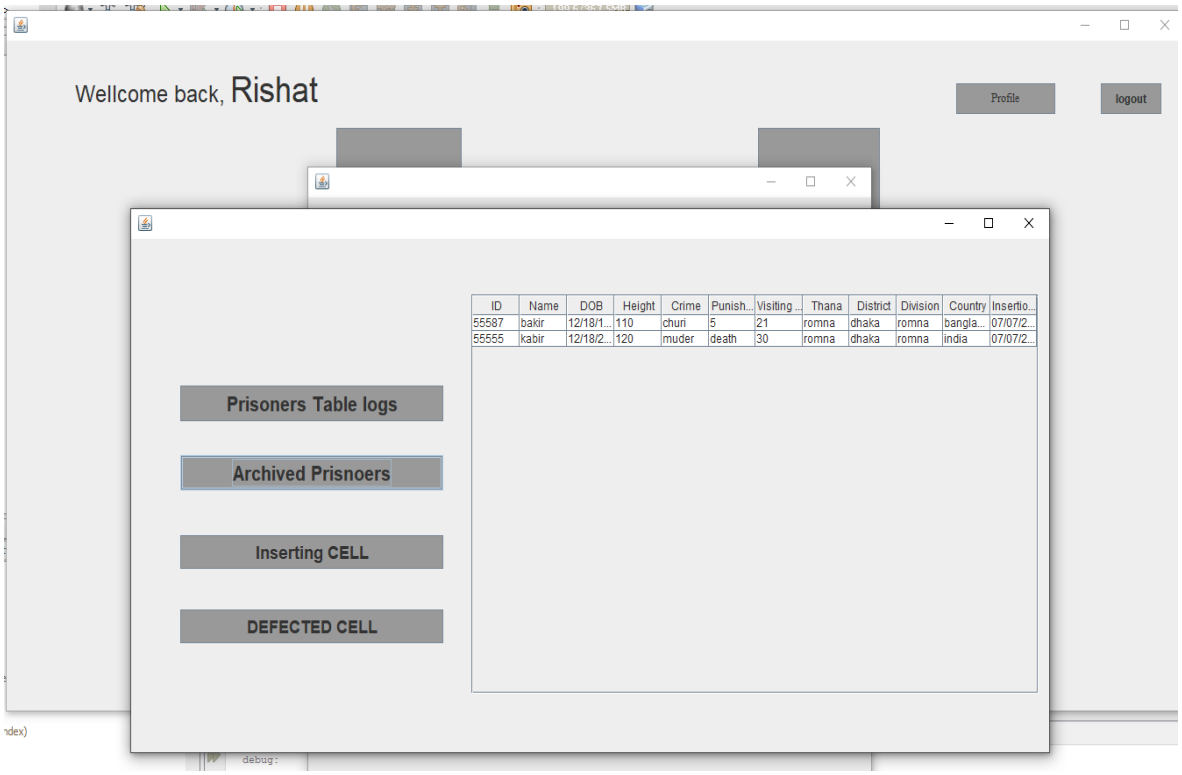

Employees:
ID
Name
Designation
Hiredate
Salary
Shift
11111 oishi jailor 07/01/2020 35000 8am-10pm
22223 babu gourd 07/01/2021 15000 6pm-2am
22289 lebu gourd 07/05/2021 15000 8am-10pm
22299 tolu gourd 5/06/2019 20000 8am-10pm
2239 molu gourd 5/06/2017 25000 8am-10pm
22041 golu gourd 5/06/2000 25000 6am-5pm
Double click to select a employee from table to mana
ID
Filter
Cancel
Id: 22041
Name: golu
CHANGE SHIFT
UPDATE SALARY
DELETE

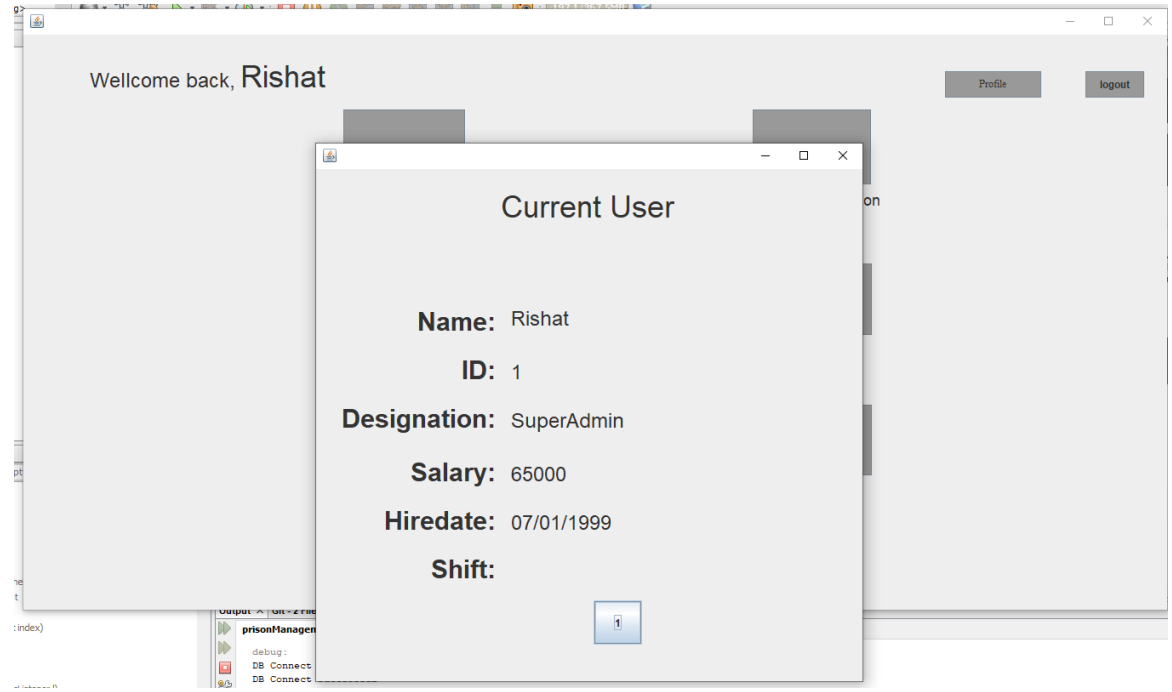

Employees:
ID
Name
Designation
11111
oishi
jailor
22223
babu
gourd
22289
lebu
gourd
22299
tolu
gourd
2239
molu
gourd
22041
golu
gourd
Double click to select a employee from table to manage.
ID
Name
Designation
Insert Employee:
Name:
Designation:
Salary:
Shift:
INSERT


Wellcome back, oishi
Profile
logout
Insert Prisoner
Prisoner Information
Manage Prisoner
Log Out
Confirm to log out?
Yes
No
Manage Cell
Mange Guard
Employees
Visitors
Super Admin
Filter


Wellcome back, Rishat
Profile
logout
Insert Prisoner
Prisoners Table logs
Archived Prisnoers
Inserting CELL
DEFECTED CELL
Cell No
Row No
Column
Floor
Building No
Capacity
Prisoners
51 2 4 105 45 2 3
50 2 4 105 45 5 1


Wellcome back, Rishat
Profile
logout
Prisoners Table logs
Archived Prisnoers
Inserting CELL
DEFECTED CELL
Inser cell's
Cell Number
Row Number
Column Number
Floor
Building Number
Capacity
Nimber of Prisoners
Guard Id
Inser

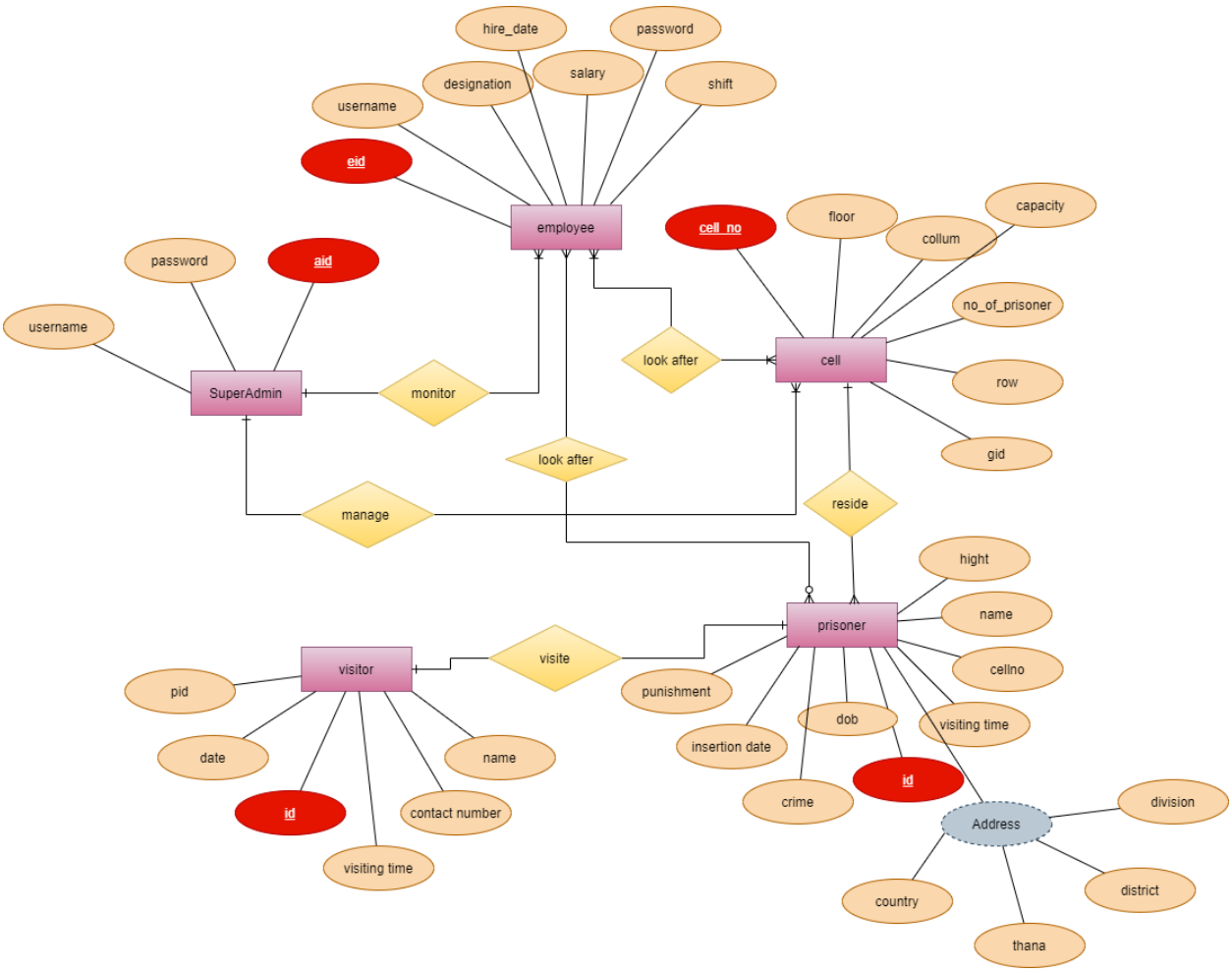

Wellcome back, Rishat
Profile
logout
ID
Name
DOB
Height
Crime
Punish...
Visiting ...
Thana
District
Division
Country
Insertio...
55587
bakir
12/18/1...
110
churi
5
21
romna
dhaka
romna
bangla...
07/07/2...
55555
kabir
12/18/2...
120
muder
death
30
romna
dhaka
romna
india
07/07/2...
Prisoners Table logs
Archived Prisnoers
Inserting CELL
DEFECTED CELL


Wellcome back, Rishat
Profile
logout
Current User
Name: Rishat
ID: 1
Designation: SuperAdmin
Salary: 65000
Hiredate: 07/01/1999
Shift:

5. Scenario Description:

In a prison management system there is employees, prisoners, cells, visitors, and super admin and every employee has their username, designation, hire date, salary, shift, password, eid and a unique id. Many employees can be monitored by one superadmin. Super admin has his own aid, password, username. A Super admin can also manage many cells. Cell has cell_no, floor, collum, capacity, no_of_prisoner, row, and gid. Many cells can be looked after by many employees, and also one cell can reside more than one prisoner. every Prisoner has his/her own height, name, cell_no, visiting_time, date_of_birth, crime, insertion_date, punishment and uniquid information. Prisoner has an address which have its attribute called division, district, thana and country. In prison management system, visitors who have their information like name, pid, date, uniquid, visiting time and contact number and one visitor can visit only one prisoner at a time.

6. ER Diagram:



7.Normalization:

Primary key

FOREIGN KEY

Monitor :

1st NF:

username ,password , aid , id, username, designation , password , salary, shift, hire_date

2nd NF:

1. username ,password , aid
2. id, username, designation , password , salary, shift, hire_date.

3rd NF

1. aid,Username , password
2. aid,id
3. id,username password ,
4. designation ,salary,shift, hire_date

Table creation :

1. username,password ,sid,rid
2. sid,aid,id
3. username, password,rid
4. rid.designation ,salary,shift, hire_date

Look after :

1st NF:

1. id, username, designation , password , salary, shift, hire_date, cellno, floor, collum, capacity, no_of_prisoners, row, gid

2nd NF:

1. id, username, designation , password , salary, shift, hire_date
2. cellno, floor, collum, capacity, no_of_prisoners, row, gid

3rd NF:

1. id
2. username, designation , password , salary, shift, hire_date
3. gid
4. cellno, floor, collum, capacity, no_of_prisoners, row

Table creation :

1. eid.sid.gid
1. sid.username,designation , password , salary, shift, hire_date
2. gid.rid.
3. rid. cellno, floor, collum, capacity, no_of_prisoners, row

Look after :

1st NF

1. height ,name ,visiting_time,dob,id,crime, insert_date, punishment , country, thana, district, division, id, username, designation , password , salary, shift, hire_date.

2nd NF:

1. height ,name ,visiting_time,dob, id, crime, insert_date, punishment , country, thana, district, division
2. id, username, designation , password , salary, shift, hire_date.

3rd NF:

1. height ,name ,visiting_time,dob, id, crime, insert_date, punishment , country, thana, district, division
2. id, username, designation , password , salary, shift, hire_date.

Table creation :

1. height ,name ,visiting_time,dob,id,crime, insert_date, punishment , country, thana, district, division
2. id, username, designation , password , salary, shift, hire_date.

Reside :

1st NF:

1. height ,name ,visiting_time,dob,id,crime, insert_date, punishment , country, thana, district, division, cellno, floor, collum, capacity, no_of_prisoners, row, gid.

2nd NF:

1. height ,name ,visiting_time,dob,id,crime, insert_date, punishment , country, thana, district, division
2. cellno, floor, collum, capacity, no_of_prisoners, row, gid

3rd NF:

1. height ,name ,visiting_time,dob,id,crime, insert_date, punishment , country, thana, district, division
2. cellno, floor, collum, capacity, no_of_prisoners, row, gid

Table creation :

1. height ,name ,visiting_time,dob, id,crime, insert_date, punishment , country, thana, district, division
2. cellno, floor, collum, capacity, no_of_prisoners, row, gid

Visite :

1st NF:

1. height ,name ,visiting_time,dob, id,crime, insert_date, punishment , country, thana, district, division,pid, pname, date,id,visiting_time,contact_number, name.

2nd NF:

1. height ,name ,visiting_time, dob, id,crime, insert_date, punishment , country, thana, district, division
2. pid, pname, date, id, visiting_time,contact_number, name

3rd NF:

1. id,name, visiting_time
2. height, dob, crime, insert_date, punishment , country, thana, district, division
3. id ,name, visiting_time
4. pid, pname,date,contact_number

Table creation :

1. id,name, visiting_time.oid.pid
2. oid.height, dob, crime, insert_date, punishment , country, thana, district, division

3. id,name, visiting_time.rid
4. rid.pid, pname,date,contact_number

Manage :

1. height ,name ,visiting_time,dob, id,crime, insert_date, punishment , country, thana, district, division, username ,password , aid

2nd NF:

2. height , name ,visiting_time,dob, id,crime, insert_date, punishment , country, thana, district, division
3. username ,password , aid

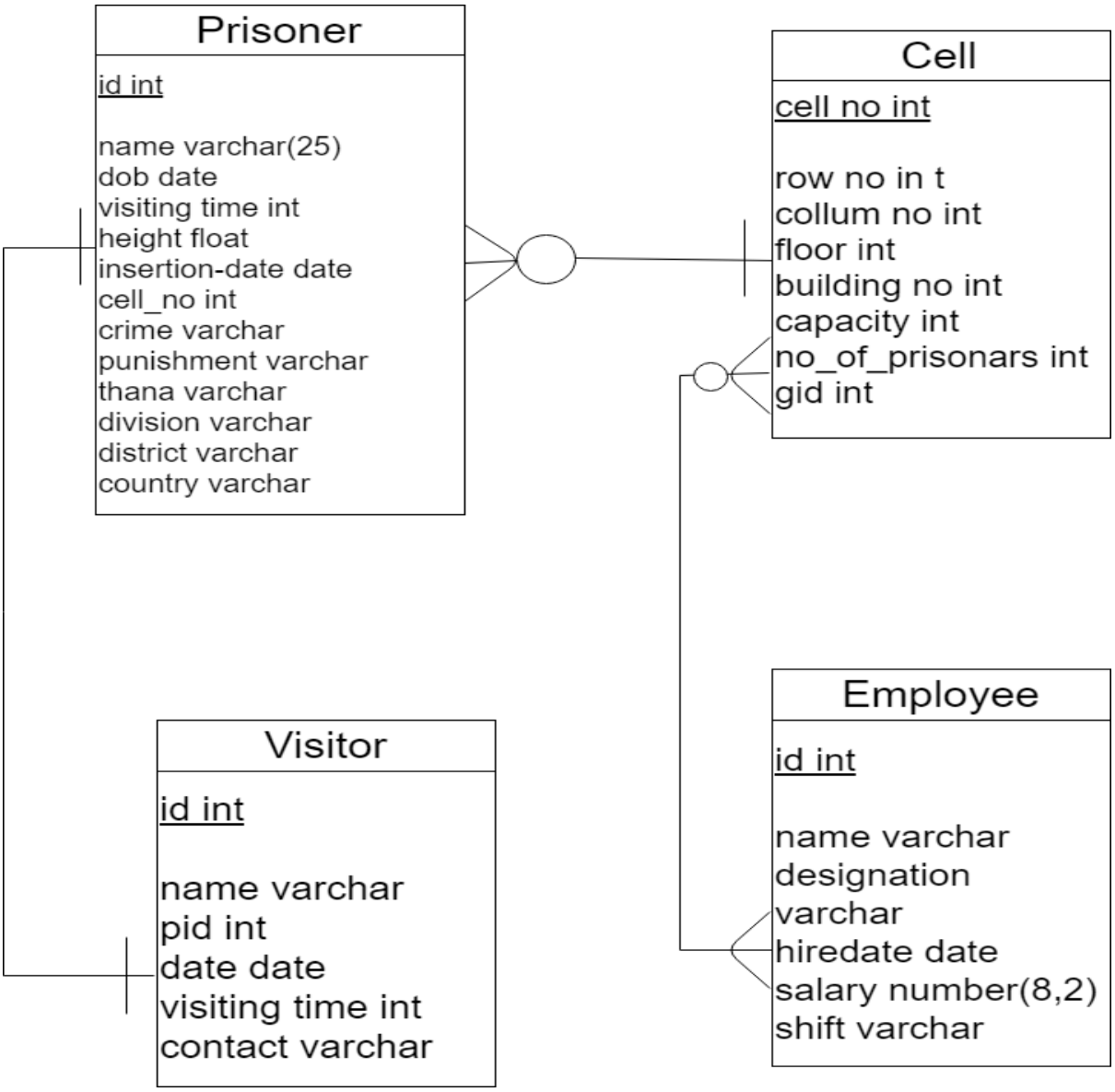
3rd NF:

1. hight ,name ,visiting_time,dob, id,crime, insert_date, punishment , country, thana, district, division
2. username ,password , aid

Table creation :

1. height ,name ,visiting_time,dob, id,crime, insert_date, punishment , country, thana, district, division
2. username ,password , aid

8. Schema Diagram:



9. Table Creation:

Employee table:

```
CREATE TABLE employee (
    id int NOT NULL,
    name varchar(255),
    designation varchar(255),
    hiredate varchar(255),
    salary varchar(255),
    shift varchar(255),
    PRIMARY KEY (id)
);
```

Admin table:

```
CREATE TABLE admin (
    username varchar(255),
   password varchar(255),
    eid int NOT NULL,
    FOREIGN KEY (eid)
REFERENCES employee(id)
);
```

Super_admin table:

```
CREATE TABLE superadmin
(
    id int NOT NULL,
    name varchar(255),
    designation varchar(255),
    hiredate varchar(255),
    salary varchar(255),
     UNIQUE(name),
    PRIMARY KEY (id)
);
```

Prisoners table:

```
CREATE TABLE prisoners (
      id int  NOT NULL UNIQUE,
      name varchar(255) NOT NULL,
     dob varchar(255),
      height int,
      cellno int,
     crime varchar(255),
    punishment varchar(255),
    visiting_time int,
     thana varchar(255),
    district varchar(255),
     division varchar(255),
    country varchar(255),
    insertion_date varchar(255),
    PRIMARY KEY (name),
    FOREIGN KEY (cellno)
 REFERENCES cell(cellno)
);
```

Visitor table:

```
CREATE TABLE visitor (
    id int NOT NULL,
    name varchar(255),
    visiting_date varchar(255),
    visiting_time varchar(255),
    contact_no int,
    pid int ,
    pname varchar(255),
    cellno int,
    PRIMARY KEY (id),
    FOREIGN KEY (Pid)
REFERENCES prisoners(id),
   FOREIGN KEY (Pname)
REFERENCES prisoners
(name),
   FOREIGN KEY (cellno)
REFERENCES cell(cellno)
```

Cell Table:

```
CREATE TABLE cell
( cellno int ,
    row_no int,
    collumn int,
    floor int,
    bulding_no int,
    capacity int,
    no_of_prisoners int,
    gid int,
 PRIMARY KEY (cellno),
   FOREIGN KEY ( gid)
REFERENCES
employee(id)
);
```

Index for employee table

CREATE INDEX employeeinformation ON employee
(id,name,designation,hiredate,salary,shift);

Index for admin table:

CREATE INDEX admininformation ON admin (eid,username,password);

Index for prisoners table

CREATE INDEX prisonersinfromtion ON prisoners (id,name,dob, height,cellno, crime, punishment,visiting_time,thana,district,division,country,insertion_date);

Index for visitor table

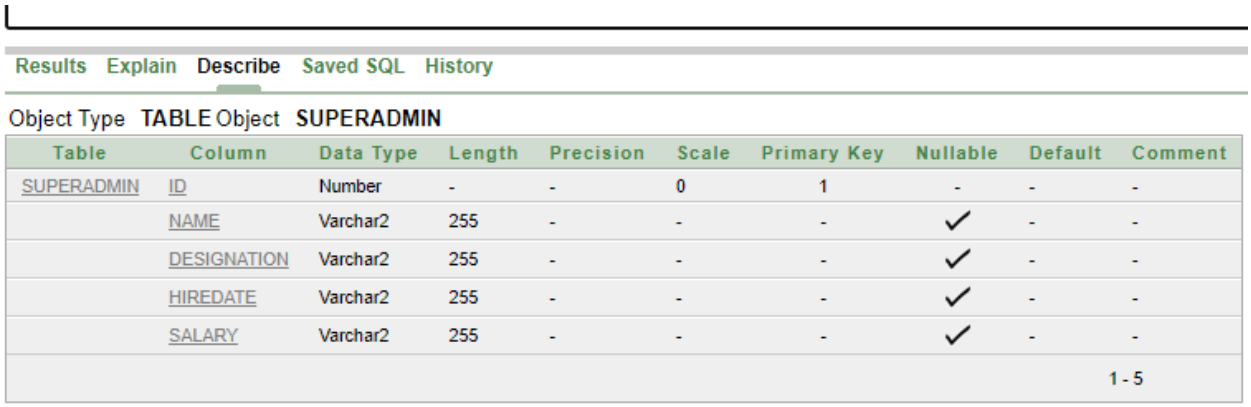
CREATE INDEX visitorinfromtion ON visitor
(id,name,visiting_date,visiting_time,contact_no,pid,pname,cellno);

Index for cell table

CREATE INDEX Cellinfromtion ON cell
(cellno,row_no,collumn,floor,bulding_no,capacity,no_of_prisoners,gid_prisoners);

The created table using describe command:

Superadmin:



Results Explain Describe Saved SQL History

Object Type TABLE Object SUPERADMIN

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
SUPERADMIN	ID	Number	-	-	0	1	-	-	-
	NAME	Varchar2	255	-	-	-	✓	-	-
	DESIGNATION	Varchar2	255	-	-	-	✓	-	-
	HIREDATE	Varchar2	255	-	-	-	✓	-	-
	SALARY	Varchar2	255	-	-	-	✓	-	-

1 - 5

Cell

Results Explain Describe Saved SQL History

Object Type TABLE Object CELL

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CELL	CELLNO	Number	-	-	0	1	-	-	-
	ROW_NO	Number	-	-	0	-	✓	-	-
	COLLUMN	Number	-	-	0	-	✓	-	-
	FLOOR	Number	-	-	0	-	✓	-	-
	BULDING_NO	Number	-	-	0	-	✓	-	-
	CAPACITY	Number	-	-	0	-	✓	-	-
	NO_OF_PRISONERS	Number	-	-	0	-	✓	-	-
	GID	Number	-	-	0	-	✓	-	-

1 - 8

Visitor

Results Explain Describe Saved SQL History

Object Type TABLE Object VISITOR

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
VISITOR	ID	Number	-	-	0	1	-	-	-
	NAME	Varchar2	255	-	-	-	✓	-	-
	VISITING_DATE	Varchar2	255	-	-	-	✓	-	-
	VISITING_TIME	Varchar2	255	-	-	-	✓	-	-
	CONTACT_NO	Number	-	-	0	-	✓	-	-
	PID	Number	-	-	0	-	✓	-	-
	PNAME	Varchar2	255	-	-	-	✓	-	-
	CELLNO	Number	-	-	0	-	✓	-	-

1 - 8

Prisoners

Results Explain Describe Saved SQL History

Object Type TABLE Object PRISONERS

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
PRISONERS	ID	Number	-	-	0	-	-	-	-
	NAME	Varchar2	255	-	-	1	-	-	-
	DOB	Varchar2	255	-	-	-	✓	-	-
	HEIGHT	Number	-	-	0	-	✓	-	-
	CELLNO	Number	-	-	0	-	✓	-	-
	CRIME	Varchar2	255	-	-	-	✓	-	-
	PUNISHMENT	Varchar2	255	-	-	-	✓	-	-
	VISITING_TIME	Number	-	-	0	-	✓	-	-
	THANA	Varchar2	255	-	-	-	✓	-	-
	DISTRICT	Varchar2	255	-	-	-	✓	-	-
	DIVISION	Varchar2	255	-	-	-	✓	-	-
	COUNTRY	Varchar2	255	-	-	-	✓	-	-
	INSERTION_DATE	Varchar2	255	-	-	-	✓	-	-

1 - 13

Employee

Results Explain Describe Saved SQL History

Object Type TABLE Object EMPLOYEE

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMPLOYEE	ID	Number	-	-	0	1	-	-	-
	NAME	Varchar2	255	-	-	-	✓	-	-
	DESIGNATION	Varchar2	255	-	-	-	✓	-	-
	HIREDATE	Varchar2	255	-	-	-	✓	-	-
	SALARY	Varchar2	255	-	-	-	✓	-	-
	SHIFT	Varchar2	255	-	-	-	✓	-	-

1 - 6

Admin

Results Explain Describe Saved SQL History

Object Type TABLE Object ADMIN

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ADMIN	USERNAME	Varchar2	255	-	-	-	✓	-	-
	PASSWORD	Varchar2	255	-	-	-	✓	-	-
	EID	Number	-	-	0	-	-	-	-

1 - 3

Create Sequence:

```
create sequence id_prisoner
start with 55560
increment by 1
maxvalue 60000
nocycle
cache 10;
```

```
create sequence id_visitor
start with 40000
increment by 1
maxvalue 55550
nocycle
cache 10;
```

```
create sequence id_employee
start with 99993
increment by 1
maxvalue 101111
nocycle
cache 10;
```

Create Role:

```
CREATE ROLE Jailor
CREATE ROLE Superadmin_
```

Give grant :

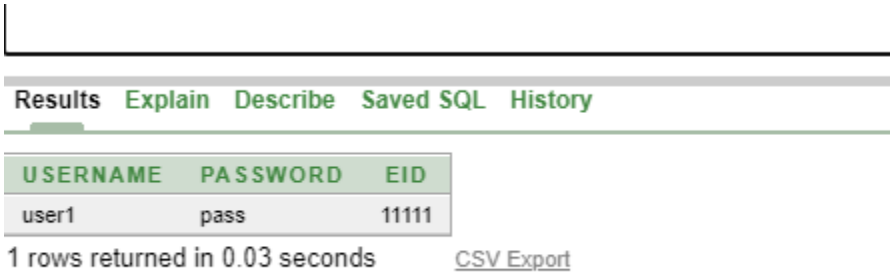
```
GRANT SELECT,UPDATE ON employee TO oishi

GRANT create table, create view to superadmin_

GRANT superadmin_ to Rishat
```

10.Data Insertion:

Admin table:



Employee table :

Results Explain Describe Saved SQL History

ID	NAME	DESIGNATION	HIREDATE	SALARY	SHIFT
11111	oishi	jailor	07/01/2020	35000	8am-10pm
2205	shumit	goard	07/01/2021	15000	8am-10pm
2250	gulu	guard	12/01/2019	25000	8am-10pm
2258	molu	guard	5/01/2011	26000	8am-10pm
2266	tolu	guard	5/05/2010	25000	8am-10pm

5 rows returned in 0.02 seconds CSV Export

Prisoners table:

Results Explain Describe Saved SQL History

ID	NAME	DOB	HEIGHT	CELLNO	CRIME	PUNISHMENT	VISITING_TIME	THANA	DISTRICT	DIVISION	COUNTRY	INSERTION_D
55555	kabir	12/18/2000	120	50	muder	death	30	romna	dhaka	romna	banglaesh	07/07/2015
55587	bakir	12/18/1998	110	65	churi	5	21	romna	dhaka	romna	banglaesh	07/07/2000
55856	barek	2/8/2003	120	50	churi	5	30	romna	dhaka	romna	banglaesh	07/07/2020
69357	maruf	9/2/2003	120	65	churi	2	36	romna	dhaka	mirpur	banglaesh	07/12/2021
63895	sakib	9/2/2013	120	23	churi	1	40	romna	dhaka	mirpur	banglaesh	07/2/2021
9685	rakib	9/2/2013	120	60	churi	4	30	pollobi	dhaka	mirpur	banglaesh	07/2/2020

6 rows returned in 0.00 seconds CSV Export

Cells table:

Results Explain Describe Saved SQL History

CELLNO	ROW_NO	COLLUMN	FLOOR	BULDING_NO	CAPACITY	NO_OF_PRISONERS	GID_PRISONERS
65	2	4	105	45	53	23	55555
50	2	4	105	45	53	23	55555
555	55	5	65	66	44	65	1255
51	5	9	10	4	50	3	69384
31	50	10	12	8	60	39	85284
23	59	15	16	19	62	69	9999
60	65	15	22	4	3	65	98299

7 rows returned in 0.00 seconds CSV Export

Insert in employee table :

INSERT INTO employee (id,name,designation,hiredate,salary,shift) VALUES ('11111','oishi','jailor','07/01/2020','35000','8am-10pm');
INSERT INTO employee (id,name,designation,hiredate,salary,shift) VALUES ('22223','babu','Guard','07/01/2021','15000','8am-10pm');

Insert in admin :

INSERT INTO admin (username, password, eid) VALUES ('user1', 'pass','11111');

Insert in cell :

insert into cell(cellno,row_no,collumn,floor,bulding_no,capacity,no_of_prisoners,gid) Values ('54','2','4','105',45,5,3,2250)
insert into cell(cellno,row_no,collumn,floor,bulding_no,capacity,no_of_prisoners,gid_prisoners) Values ('50','2','4','105',45,53,23,55555)

insert into cell(cellno,row_no,collumn,floor,bulding_no,capacity,no_of_prisoners,gid) Values ('54','2','4','105',45,2,3,22223)

Insert in prisoners :

insert into prisoners (id,name,dob,height,cellno,crime,punishment,visiting_time, thana,district,division,country,insertion_date)values('55555','kabir','12/18/2000','120','50','muder', 'death','30','romna','dhaka','romna','banglaesh','07/07/2015');

insert into prisoners (id,name,dob,height,cellno,crime,punishment,visiting_time, thana,district,division,country,insertion_date)values('55587','bakir','12/18/1998','110','65','churi',' 5','21','romna','dhaka','romna','banglaesh','07/07/2000');

insert into prisoners (id,name,dob,height,cellno,crime,punishment,visiting_time, thana,district,division,country,insertion_date)values('55555','kabir','12/18/2000','120','54','muder', 'death','30','romna','dhaka','romna','india','07/07/2015');

Insert Visitor:

insert into visitor (id,name, visiting_date,visiting_time,contact_no,pid,pname,cellno)

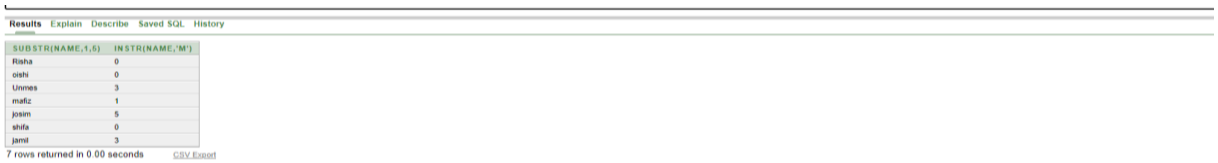
values('53569','bakir','12/18/1998','30','01814456215','55555','kabir','54');

11. Query Writing:

3 single-row function –

1.Write a query to returns the portion of input string from 1st position to 5th position and returns the numeric position of character 'm' in the employee name?

- SELECT SUBSTR (name,1,5), INSTR (name,'m') FROM employee WHERE rownum < 10;



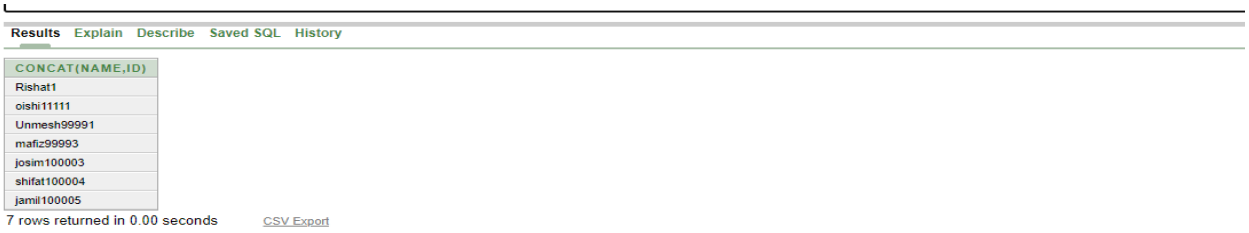
Results Explain Describe Saved SQL History

SUBSTR(NAME,1,5)	INSTR(NAME,'M')
Risha	0
oishi	0
Unmes	3
mafiz	1
josim	5
shifa	0
jamil	3

7 rows returned in 0.00 seconds CSV Export

2.Write a query a SELECT query below demonstrates the use of CONCAT function to concatenate two values.?

- SELECT CONCAT (name,id) FROM employee WHERE rownum < 10;



Results Explain Describe Saved SQL History

CONCAT(NAME,ID)
Rishat1
oishi11111
Unmesh99991
mafiz99993
josim100003
shifat100004
jamil100005

7 rows returned in 0.00 seconds CSV Export

3.Write a query to List out the foreign prisoners.

-select * from prisoners where upper(country) not in ('bd');

Results Explain Describe Saved SQL History

ID	NAME	DOB	HEIGHT	CELLNO	CRIME	PUNISHMENT	VISITING_TIME	THANA	DISTRICT	DIVISION	COUNTRY	INSERTION_DATE
9685	rakib	9/2/2013	120	60	churi	4	30	pollobi	dhaka	mirpur	banglaesh	07/2/2020
55555	kabir	12/18/2000	120	50	muder	death	30	romna	dhaka	romna	banglaesh	07/07/2015
55587	bakir	12/18/1998	110	65	churi	5	21	romna	dhaka	romna	banglaesh	07/07/2000
55856	barek	2/8/2003	120	50	churi	5	30	romna	dhaka	romna	banglaesh	07/07/2020
63895	sakib	9/2/2013	120	23	churi	1	40	romna	dhaka	mirpur	banglaesh	07/2/2021
69357	maruf	9/2/2003	120	65	churi	2	36	romna	dhaka	mirpur	banglaesh	07/12/2021

6 rows returned in 0.00 seconds CSV Export

3 group function –

[1]Write a query to displays number of employees work in each department.

-SELECT designation, COUNT (*) FROM employee GROUP BY designation;

Results Explain Describe Saved SQL History

DESIGNATION	COUNT(*)
gourd	5
jailor	1

2 rows returned in 0.00 seconds CSV Export

[2] Write a query to displays number of employees, total salary paid to employees work in each department.

- SELECT designation "Department Code", COUNT(*) "No of Employees", SUM(salary) "Total Salary" FROM employee GROUP BY designation;

Results Explain Describe Saved SQL History

Department Code	No Of Employees	Total Salary
gourd	5	100000
jailor	1	35000

2 rows returned in 0.00 seconds CSV Export

[3] Write a query to displays the cellno, number of cell of those groups that have more than 6 no of prisoner?

-SELECT cellno, count(*) "No. of Employee" FROM cell GROUP BY cellno HAVING count(*)< 6;

Results Explain Describe Saved SQL History

CELLNO	No. Of Employee
23	1
31	1
50	1
51	1
60	1
65	1
555	1

7 rows returned in 0.00 seconds CSV Export

3 subquery :

1. Show the employee their salary less than 3000?

   Ans: select * from employee where id in (select id from employee where salary <30000);

Results Explain Describe Saved SQL History

EID	NAME	DESIGNATION	HIREDATE	SALARY	SHIFT
2239	molu	gourd	5/06/2017	25000	8am-10pm
22041	golu	gourd	5/06/2000	25000	6am-5pm
22223	babu	gourd	07/01/2021	15000	6pm-2am
22289	lebu	gourd	07/05/2021	15000	8am-10pm
22299	tolu	gourd	5/06/2019	20000	8am-10pm

5 rows returned in 0.02 seconds CSV Export

2. Increase the salary of those employees, whose salary is less than or equal 1500?

   Ans : update employee set salary = salary * 0.25 where salary in (select salary from employee where salary >= 15000);

Results Explain Describe Saved SQL History

```
6 row(s) updated.

0.01 seconds
```

3. Write a query to find out the guard who joined last?

   Ans : select * from employee where designation='Guard'
   and hiredate=(Select max(hiredate) from employee where designation='Guard');

Results Explain Describe Saved SQL History

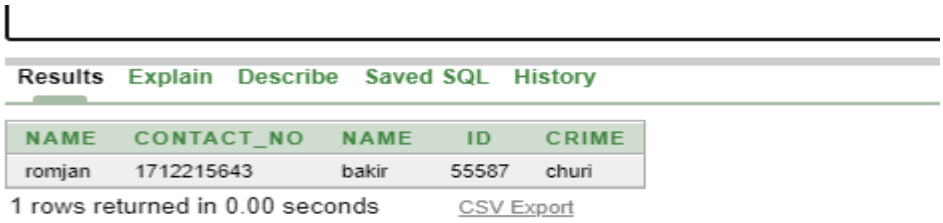
ID	NAME	DESIGNATION	HIREDATE	SALARY	SHIFT
100003	josim	Guard	13-AUG-21	4511	8am-10pm
100004	shifat	Guard	13-AUG-21	2400	10am-6pm

2 rows returned in 0.01 seconds CSV Export

3 joining –

Write a query to show the guard ?

1. Write a query to show the visitors who visit the prisoner?
   - SELECT visitor.name,visitor.contact_no,prisoners.name,prisoners.id, prisoners.crime
   FROM visitor INNER JOIN prisoners ON visitor.pid = prisoners.ID;

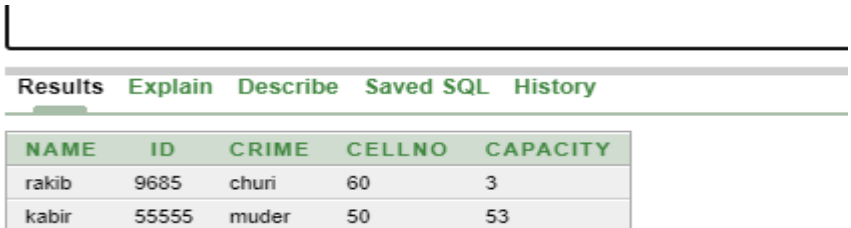


Results Explain Describe Saved SQL History

NAME	CONTACT_NO	NAME	ID	CRIME
romjan	1712215643	bakir	55587	churi

1 rows returned in 0.00 seconds CSV Export

2. Write a query to show the cell details ?
   - SELECT prisoners.name,prisoners.id, prisoners.crime,cell.cellno,cell.capacity
   FROM prisoners
   left JOIN cell ON prisoners.cellno=cell.cellno;



Results Explain Describe Saved SQL History

NAME	ID	CRIME	CELLNO	CAPACITY
rakib	9685	churi	60	3
kabir	55555	muder	50	53
bakir	55587	churi	65	53
barek	55856	churi	50	53
sakib	63895	churi	23	62
maruf	69357	churi	65	53

6 rows returned in 0.00 seconds CSV Export

3. Write a query to show the cell number of prisoners?
   - SELECT prisoners.name,prisoners.id, prisoners.crime,cell.cellno
   FROM prisoners INNER JOIN cell ON prisoners.id = cell.gid_prisoners;

Results Explain Describe Saved SQL History

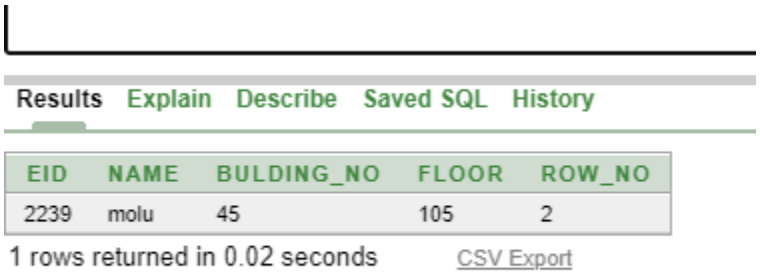
NAME	ID	CRIME	CELLNO
kabir	55555	muder	50
kabir	55555	muder	65

2 rows returned in 0.00 seconds CSV Export

3 view query:

1.Create a view of the total number of prisoners brought into in 2017?

Ans : CREATE VIEW [Total prisoner into 2017] AS SELECT count(*) FROM prisoners WHERE to_char(insertion_date,'YYYY')='2017';



Results Explain Describe Saved SQL History

EID	NAME	BULDING_NO	FLOOR	ROW_NO
2239	molu	45	105	2

1 rows returned in 0.02 seconds CSV Export

1. Create a view of the salary for find out min max and average salary …

Ans: create or replace view salary as select sum(salary) "Sum_of_Salary",avg(salary) "Average_Salary",max(salary) "Maximum_Salary",min(salary) "Minimum_Salary",designation;

Results Explain Describe Saved SQL History

Sum_of_Salary	Average_Salary	Maximum_Salary	Minimum_Salary	DESIGNATION
100000	20000	25000	15000	gourd
35000	35000	35000	35000	jailor

2 rows returned in 0.14 seconds CSV Export

3. create a view to show the visitors ?

- create or replace view visitor_prisoner as select v.id,v.name, v.visiting_date, v.visiting_time,contact_no,v.pid,p.name "PNAME",p.cellno from visitor v,prisoners p where v.pid=p.id;

Results Explain Describe Saved SQL History

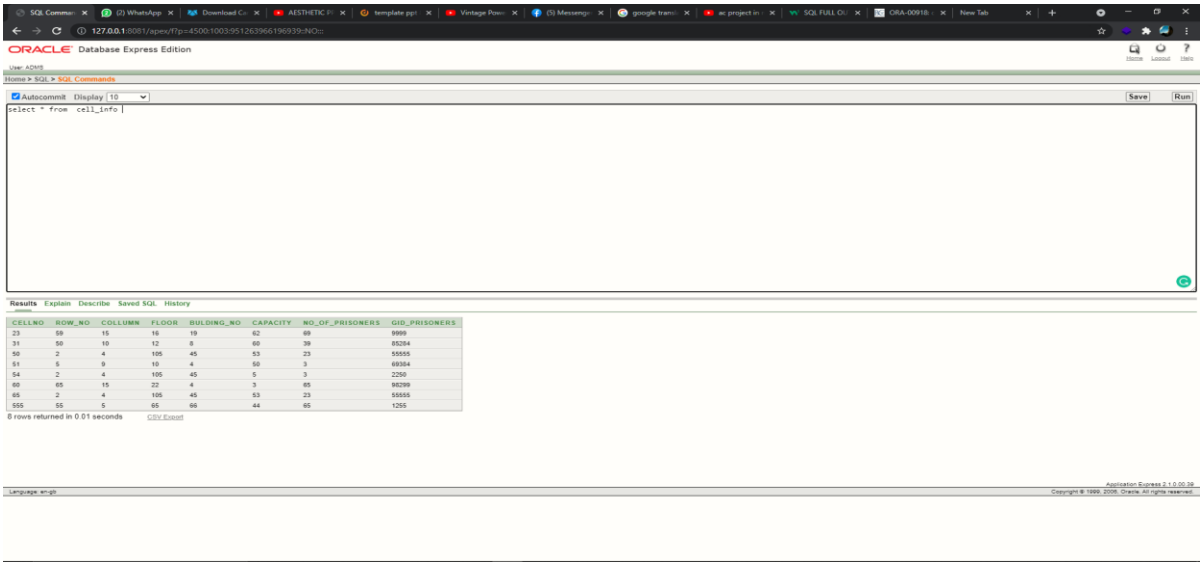
ID	NAME	VISITING_DATE	VISITING_TIME	CONTACT_NO	PID	PNAME	CELLNO
5	romjan	4/8.2021	30	1712215643	55587	bakir	65

1 rows returned in 0.01 seconds CSV Export

3 synonym:

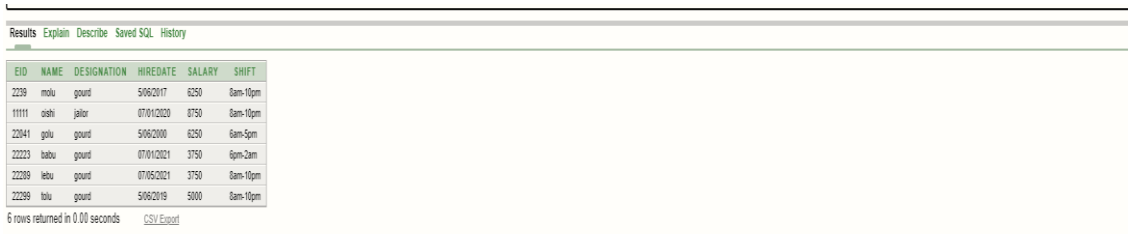
1. Create a synonym from cell ?

- CREATE SYNONYM cell_number FOR adms.cell;



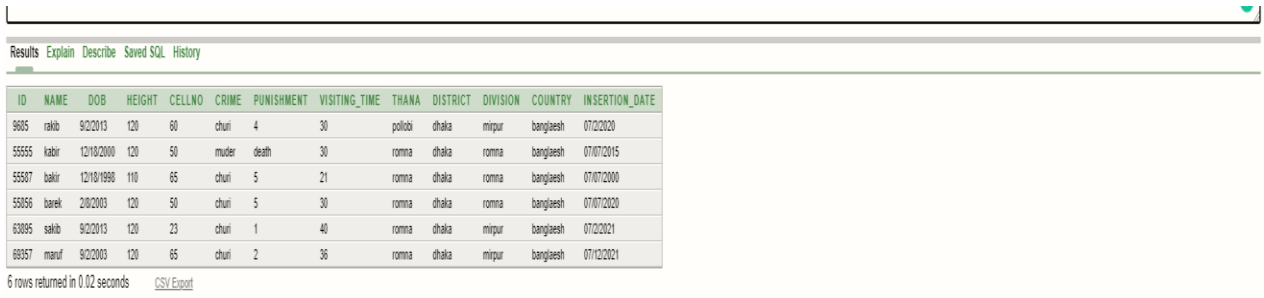
2. Create a synonym from employee?

- CREATE SYNONYM emp_info FOR adms.employe;



3.Create a synonym from employee?

-CREATE SYNONYM prisoners_info FOR adms.prisoners;

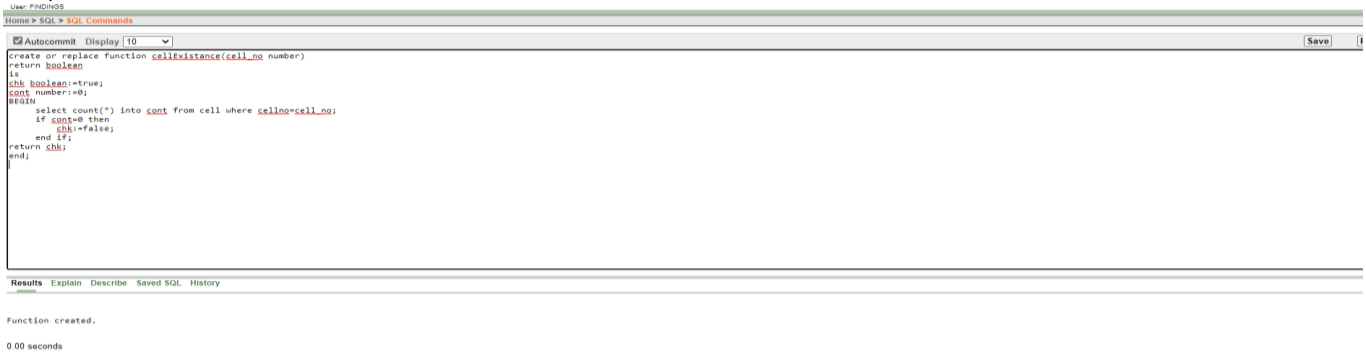


PL/SQL

-3 function

[1]Write a PL/SQL function to find out the cell existence in this thana.

```
create or replace function cellExistance(cell_no number)
return boolean
is
chk boolean:=true;
cont number:=0;
BEGIN
    select count(*) into cont from cell where cellno=cell_no;
    if cont=0 then
       chk:=false;
    end if;
return chk;
end;
```



[2]Write a PL/SQL function to find out which cell's are empty.

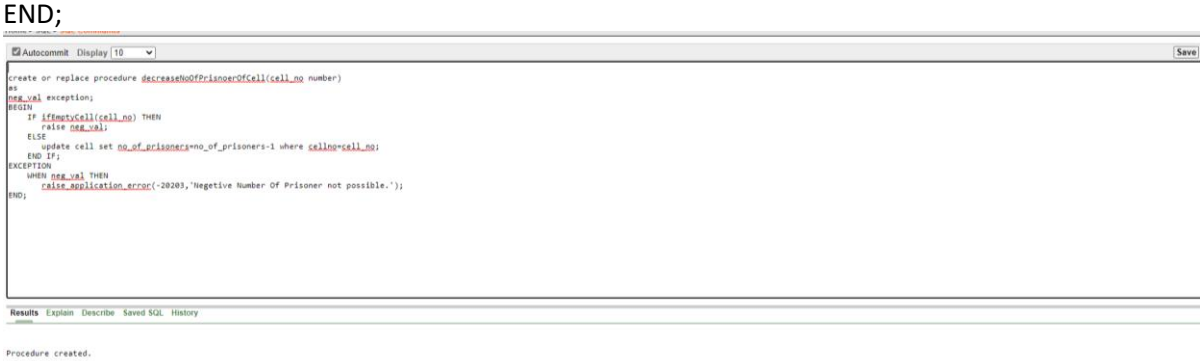
```
create or replace function ifEmptyCell(cell_no number)
return boolean
is
noOfPrisoner number;
chk boolean:=false;
BEGIN
   select no_of_prisoners into noOfPrisoner from cell where cellno=cell_no;
   IF noOfPrisoner=0 THEN
      chk:=true;
   END IF;
return chk;
END;
```

[3] Write a PL/SQL function to find out the total cell

-3 procedure

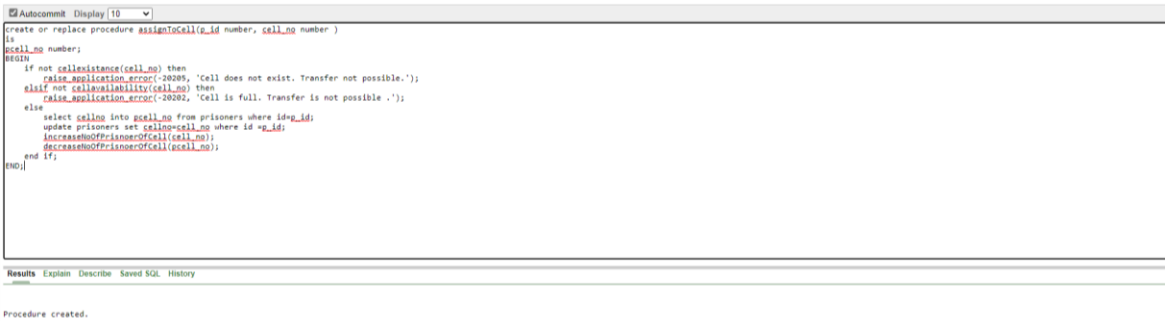
[1] Write a PL/SQL procedure to decrease the number of prisoner of a cell.

```
create or replace procedure decreaseNoOfPrisnoerOfCell(cell_no number)
as
neg_val exception;
BEGIN
   IF ifEmptyCell(cell_no) THEN
     raise neg_val;
   ELSE
     update cell set no_of_prisoners=no_of_prisoners-1 where cellno=cell_no;
   END IF;
EXCEPTION
   WHEN neg_val THEN
     raise_application_error(-20203,'Negetive Number Of Prisoner not possible.');
END;
```



[2]Write a PL/SQL function to find out the assign cell for the prinosers.

```
create or replace procedure assignToCell(p_id number, cell_no number )
is
pcell_no number;
BEGIN
   if not cellexistance(cell_no) then
      raise_application_error(-20205, 'Cell does not exist. Transfer not possible.');
   elsif not cellavailability(cell_no) then
      raise_application_error(-20202, 'Cell is full. Transfer is not possible .');
   else
      select cellno into pcell_no from prisoners where id=p_id;
      update prisoners set cellno=cell_no where id =p_id;
      increaseNoOfPrisnoerOfCell(cell_no);
      decreaseNoOfPrisnoerOfCell(pcell_no);
   end if;
END;
```



[3]Write a PL/SQL function to find out the prisoner Of Cell.

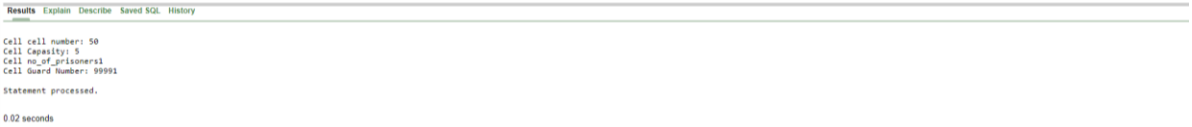
```
create or replace procedure prisonerOfCell(cell number,p1 out number,p2 out number)
is
   i number;
   j number:=0;
   cont number(1):=0;
   cursor c is
      select id from prisoners where cellno=cell;
BEGIN
   p1:=0;
   p2:=0;
   select count(*) into cont from prisoners where cellno=cell;
   IF cont=1 THEN
      FOR i in c loop
         p1:=i.id;
      END LOOP;
   ELSIF cont=2 THEN
      FOR i in c loop
         p2:=i.id;
         IF j=1 THEN
```

```
            EXIT;
          END IF;
          p1:=i.id;
          j:=j+1;
        END LOOP;
    END IF;
END;
```

-3 record

[1] Write a PL/SQL query to print the information of cell number 50.

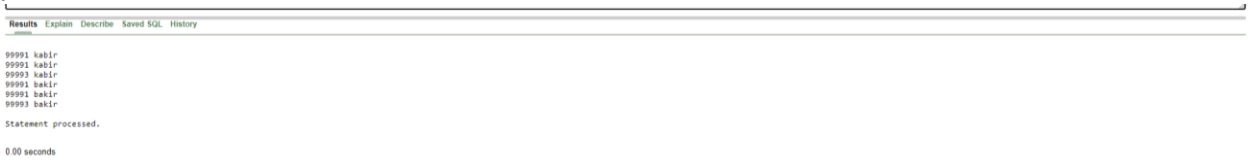
```
DECLARE
  cell_rec cell%rowtype;
BEGIN
SELECT * into cell_rec FROM cell
 WHERE cellno = 50;
  dbms_output.put_line('Cell cell number: ' || cell_rec.cellno);
  dbms_output.put_line('Cell Capasity: ' || cell_rec.capacity);
  dbms_output.put_line('Cell no_of_prisoners' || cell_rec.no_of_prisoners);
  dbms_output.put_line('Cell Guard Number: ' || cell_rec.gid);
END;
/
```



[2] Write a PL/SQL query to show that which prisoners under which guard.

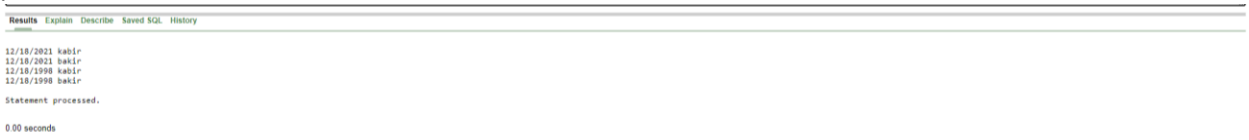
```
DECLARE
  CURSOR info_cur is
    SELECT p.id, p.name, c.gid
    FROM prisoners p,cell c;
  info_rec info_cur%rowtype;
BEGIN
```

```
  OPEN info_cur;
  LOOP
    FETCH info_cur into info_rec;
    EXIT WHEN info_cur%notfound;
    DBMS_OUTPUT.put_line(info_rec.id || ' ' || info_rec.name);
  END LOOP;
END;
/
```



[3] Write a PL/SQL query to show which prisoner have visitor in which date

```
DECLARE
  CURSOR info_cur is
    SELECT  p.name, v.visiting_date
    FROM prisoners p,visitor v;
  info_rec info_cur%rowtype;
BEGIN
  OPEN info_cur;
  LOOP
    FETCH info_cur into info_rec;
    EXIT WHEN info_cur%notfound;
    DBMS_OUTPUT.put_line(info_rec.visiting_date|| ' ' || info_rec.name);
  END LOOP;
END;
/
```



-3 cursor

[1] Write a PL/SQL query to update the salary ..

```
DECLARE
total_rows number(2);
BEGIN
UPDATE employee
SET salary = salary + 500;
IF sql%notfound THEN
dbms_output.put_line('no sal updated');
```

```
ELSIF sql%found THEN
total_rows := sql%rowcount;
dbms_output.put_line( total_rows || ' sal updated ');
END IF;
END; /
```

```
7 sal updated
Statement processed.

0.02 seconds
```

[2] write a PL/SQL query to find out the employee name's and salary

```
declare
name employee.name%type;
salary employee.salary%type;
cursor c_employee is
select name,salary from employee;
begin
open c_employee;
fetch  c_employee into name,salary;
dbms_output.put_line(name||' '||salary);
close c_employee;
end
/
```

```
Results  Explain  Describe  Saved SQL  History

Rishat 66000

Statement processed.

0.01 seconds
```

[3] Write a PL/SQL query to find out the prisoners name's and punishment.

```
Ans : DECLARE
  id PRISONERS.id%type;
  name PRISONERS.name%type;
  punishment PRISONERS.punishment%type;
CURSOR c_PRISONERS is
 SELECT id, name,punishment FROM PRISONERS;
BEGIN
  OPEN c_PRISONERS;
  LOOP
  FETCH c_prisoners into id, name, punishment;
    EXIT WHEN c_prisoners%notfound;
    dbms_output.put_line(id || ' ' || name || ' ' || punishment);
END LOOP;
CLOSE c_PRISONERS;
END;
```

```
Results  Explain  Describe  Saved SQL  History

55555 kabir death
55587 bakir 5

Statement processed.

0.01 seconds
```

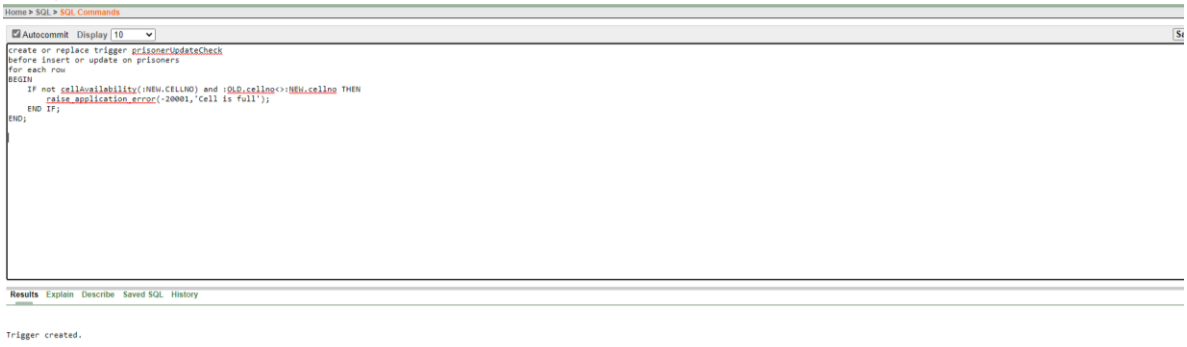
-3 trigger

[1]Write a PL/SQL trigger to archive the prisoners information.

```
create or replace trigger archivep
before delete on prisoners
for each row
BEGIN
   IF :OLD.punishment<>'Death' THEN
      IF (sysdate-:OLD.insertion_date)/365<:OLD.punishment THEN
         raise_application_error(-20209,'Archiving failed');
      END IF;
   END IF;
END;
```

[2]Write a PL/SQL trigger to check the update of the prisoners information.

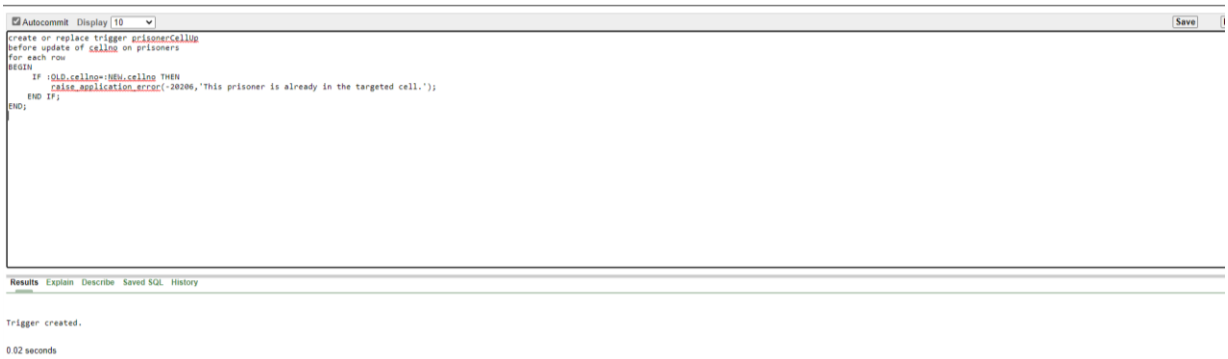
```
create or replace trigger prisonerUpdateCheck
before insert or update on prisoners
for each row
BEGIN
   IF not cellAvailability(:NEW.CELLNO) and :OLD.cellno<>:NEW.cellno THEN
      raise_application_error(-20001,'Cell is full');
   END IF;
END;
```



[3]Write a PL/SQL trigger to check the prisoners cell up.

```
create or replace trigger prisonerCellUp
before update of cellno on prisoners
for each row
BEGIN
    IF :OLD.cellno=:NEW.cellno THEN
```

```
      raise_application_error(-20206,'This prisoner is already in the targeted cell.');
   END IF;
END;
```



-3 package

[1] create a package to maintain the employee salary.

```
CREATE PACKAGE cust_sal AS
  PROCEDURE find_sal(c_id EMPLOYEE.id%type);
END cust_sal;
/
CREATE OR REPLACE PACKAGE BODY cust_sal AS
  PROCEDURE find_sal(c_id EMPLOYEE.id%TYPE) IS
  c_sal EMPLOYEE.salary%TYPE;
  BEGIN
    SELECT salary INTO c_sal
    FROM EMPLOYEE
    WHERE id = c_id;
    dbms_output.put_line('Salary: '|| c_sal);
  END find_sal;
END cust_sal;
/
```

[2]Create a package for deleting inserting counting the cell number capacity.

```
CREATE OR REPLACE PACKAGE c_package AS
 PROCEDURE addCell(
 c_cellno  cell.cellno%type,
  c_capacity cell.capacity%type,
  c_bulding_no cell.bulding_no%type,
  c_floor cell.floor%type);
   PROCEDURE delCell(c_cellno  cell.cellno%TYPE);
 PROCEDURE listCell;
```

```
 END c_package;
/
```
Body
```
CREATE OR REPLACE PACKAGE BODY c_package AS
   PROCEDURE addCell(
  c_cellno  cell.cellno%type,
   c_capacity cell.capacity%type,
   c_bulding_no cell.bulding_no%type,
   c_floor cell.floor%type
)
   IS
   BEGIN
      INSERT INTO cell(cellno, capacity ,bulding_no, floor)
         VALUES(c_cellno, c_capacity,  c_bulding_no, c_floor);
   END addCell;
    PROCEDURE delCell(c_cellno  cell.cellno%type) IS
   BEGIN
      DELETE FROM cell
      WHERE cellno = c_cellno;
   END delcell;
   PROCEDURE listcell IS
   CURSOR c_cell is
      SELECT  capacity FROM cell;
   TYPE c_list is TABLE OF cell.capacity%type;
   capacity_list c_list := c_list();
   counter integer :=0;
   BEGIN
      FOR n IN c_cell LOOP
      counter := counter +1;
      capacity_list.extend;
     capacity_list(counter) := n.capacity;
      dbms_output.put_line('Cell(' ||counter|| ')'||capacity_list(counter));
      END LOOP;
   END listCell;
   END c_package;
/
```


[3] Create a package for giving bonus to the employees.
```
CREATE or replace PACKAGE emp_bonuss AS
   PROCEDURE calc_bonus(date_hired employee.hiredate%TYPE);
END emp_bonuss;
/
CREATE or replace PACKAGE BODY emp_bonuss AS
PROCEDURE calc_bonus (date_hired employee.hiredate%TYPE) IS
   BEGIN
```

```
    DBMS_OUTPUT.PUT_LINE('Employees hired on ' || date_hired || ' get bonus.');
  END;
END emp_bonuss;
/
```

12. Conclusion:

Project findings,

The main concern of mine is to create complete prison management is to centralized the whole prison of our country or A division or a district at least. There will be no chance to lose any data of any employees or any prisoner even there will be recorded the information of a visitor who and where and when came or gone, there will be no chance to do any kind of cheating or corruption. There will be a chance to check all prison if it is empty of fill, Main goal of this project is to reduce physical work and increase the security system and proper way to manage a prison easy and proper and peaceful way.

Future work,

There are some feather that can be developed in future like, there will e head office like whole prison of Bangladesh can be control from here and yeah there will need a high secure system also , this system will be run on automatic with the help of AI and IOT, like if any cell is empty then a signal will be sent to the system that the cell is empty , and also there will be a reminder system like if any prisoner leave the cell before the release date then it will sent a signal and the system will show the report and rang the bell , on the other hand if it is a release time of any prisoner system will send a reminder message to the admin and so on ..