**RTO MANAGEMENT**

**SYSTEM**

**SUBJECT: DATABASE MANAGEMENT SYSTEM**

**DHARAMSINH DESAI UNIVERSITY**

**FACULTY OF TECHNOLOGY, NADIAD**



**PREPARED BY:** **GUIDED BY**

HARSH SHAH (IT-102) PROF. R.M.RAVAL

DISHA SHAH (IT-101) (Assistant Prof.)

**INDEX**

1. Certificate………………………………..

2. Acknowledgement………………………

3. System Overview……………

4. ER Diagram…………………………

5. Database Schema…………………………

4. Implementation………………………

* Create queries
* Insert queries
* Triggers

5. Other functionalities…………………………

6. Bibliography……………………………

**DHARAMSINH DESAI UNIVERSITY**

**DEPARTMENT OF INFORMATION TECHNOLOGY**



**CERTIFICATE**

This is to certify that **HARSH SHAH AND DISHA SHAH** student of Bachelor of Technology, Semester V (Information Technology) has completed her semester project work titled “**RTO MANAGEMENT SYSTEM**” in subject Database Management System satisfactorily in partial fulfillment of requirement during academic year 2017-2018.

**PROF. R.M. RAVAL PROF. R.S.Chhajjed**

**(Assistant Prof.) (Head of dept.)**

**ACKNOWLEDGMENT**

On completion of project we would like to express our sincere thanks to all those who have guide, advised, inspired and supported during our project work.

Every work that one completes successfully stands on the constants encouragement, good will and support of the people around. We, hereby, avail this opportunity to express our heartfelt gratitude to a number of people who extended their valuable time, full support and cooperation in developing this project.

We are heartily thankful to the qualified staff of the center and especially to our lab faculty **Prof. R.M. Raval**. We believe that her computer expertise and valuable guidance have made it possible to present such nice project report.

Thanking you.

Yours Sincerely,

**HARSH SHAH (IT - 102)**

**DISHA SHAH (IT -101)**

**SYSTEM OVERVIEW**

This is a project based on **PL/SQL,MYSQL,PHP,HTML,CSS,JAVASCRIPT** language. In this, an online RTO management system runs. The database consists of data related to:

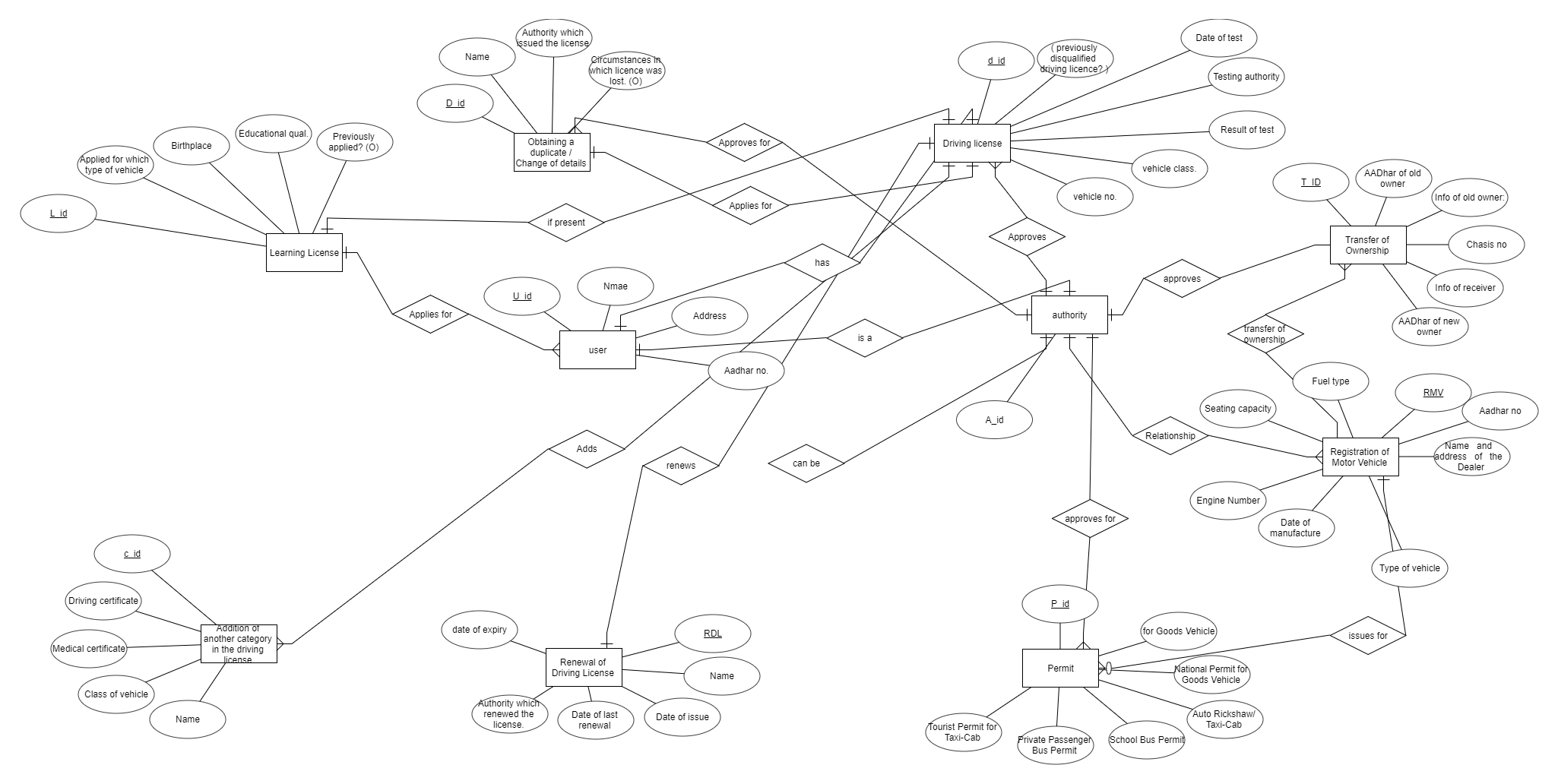
* All information about the users around the country.
* The total info about all the prisoners in those prisons.
* The types of cases on those prisoners
* The necessary info of visitors visiting the prisoners
* The work done by prisoners during their imprisonment
* The medical health of the prisoners
* The staff managing the prison

All updates done on data items are reflected on database. It contains Triggers for granting permission to visitor to visit permission, for adding prisoner into prison upto prison capacity only, updating prison capacity accordingly when a prisoner is released.

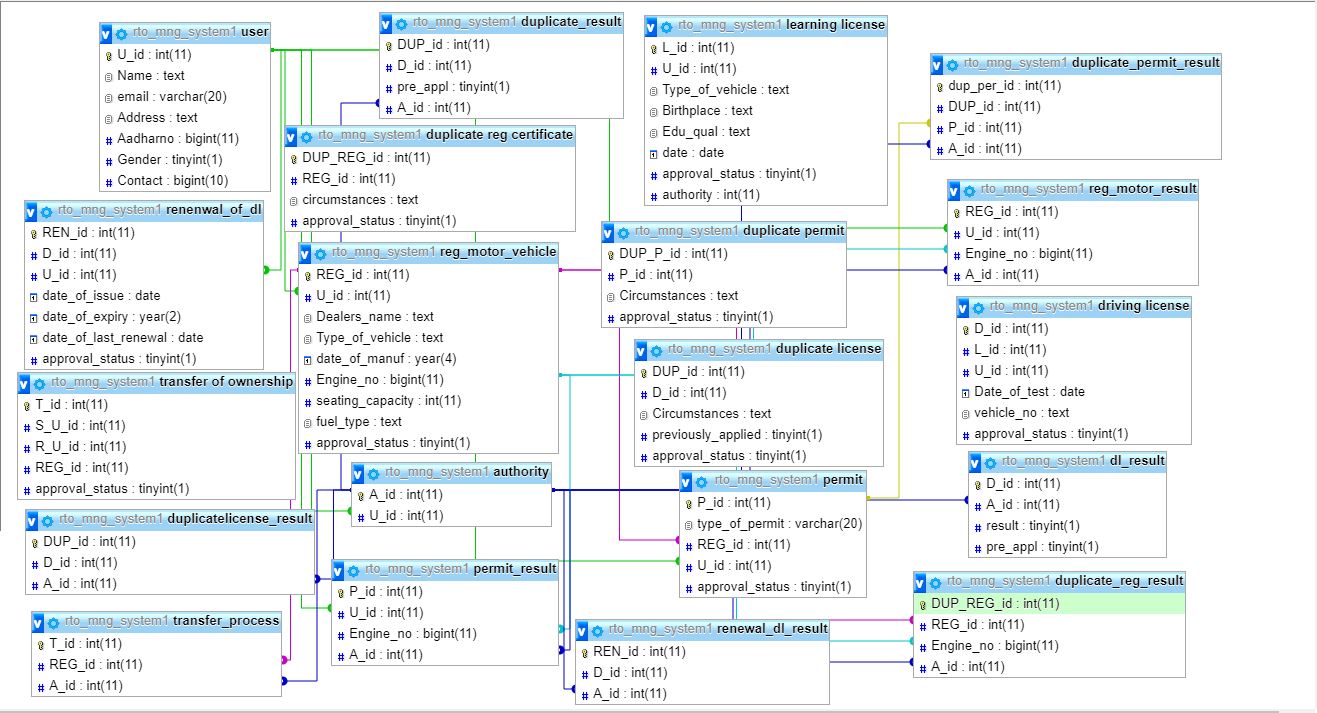
**Advantages**

The whole system is online so all data about prisons around the country and latest updates in particular prison in any corner of country is directly updated. All data related to prisoners is available to government at any hour of the day. The tedious method of going through all the files for fetching the details of any prisoner is removed, one click and all details are displayed. Any official can find data of any prison in some other place sitting at his/her place.

The main aim is that all the data is centralized thus leading to faster, easier, correct and latest updated data access.

**ER DIAGRAM**

**DATABASE SCHEMA**



**IMPLEMENTATION**

*CREATE QUERIES*

-------------------------------------**FOR TABLE AUTHORITY** ------------------------------------

Table structure for table `authority`

CREATE TABLE `authority` (

`A\_id` int(11) NOT NULL,

`U\_id` int(11) NOT NULL

)

ALTER TABLE `authority` ADD PRIMARY KEY (`A\_id`), ADD KEY `U\_id` (`U\_id`);

ALTER TABLE `authority` ADD CONSTRAINT `authority\_ibfk\_1` FOREIGN KEY (`U\_id`) REFERENCES `user` (`U\_id`);

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

---------------------------------------**FOR TABLE LEARNING LICENSE**------------------------------------

CREATE TABLE `learning license` (

`L\_id` int(11) NOT NULL, `U\_id` int(11) NOT NULL,

`Type\_of\_vehicle` text NOT NULL, `Birthplace` text NOT NULL,

`Edu\_qual` text NOT NULL, `date` date NOT NULL,

`approval\_status` tinyint(1) NOT NULL DEFAULT '0'

)

ALTER TABLE `learning license` ADD PRIMARY KEY (`L\_id`), ADD KEY `U\_id` (`U\_id`);

ALTER TABLE `learning license` ADD CONSTRAINT `learning license` FOREIGN KEY (`U\_id`) REFERENCES `user` (`U\_id`);

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

--------------------------------------------------**FOR TABLE CELL** --------------------------------------------------

CREATE TABLE CELL

(

CELL\_ID INT(10) NOT NULL AUTO\_INCREMENT,

JAIL\_ID INT(10) NOT NULL,

PRIMARY KEY (CELL\_ID),

FOREIGN KEY (JAIL\_ID) REFERENCES PRISON(JAIL\_ID) ON DELETE CASCADE ON UPDATE CASCADE

);

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

----------------------------------------------**FOR TABLE PERMISSION**------------------------------------------

CREATE TABLE PERMISSION

(

PERMISSION\_ID INT(10) NOT NULL AUTO\_INCREMENT,

USERID INT(10) NOT NULL,

RELATION VARCHAR(20) NOT NULL ,

FNAME VARCHAR(20) NOT NULL,

PRISONER\_ID INT(10) NOT NULL,

ACCESS INT(10) ,

PRIMARY KEY (PERMISSION\_ID),

FOREIGN KEY (USERID) REFERENCES USER1(USERID) ON DELETE CASCADE ON UPDATE CASCADE);

);

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

--------------------------------------------------**FOR TABLE STAFF**------------------------------------------------

CREATE TABLE STAFF

(

STID INT(20) NOT NULL AUTO\_INCREMENT,

SNAME VARCHAR(20) NOT NULL,

EMAIL VARCHAR(20) NOT NULL,

PASSWORD VARCHAR(20) NOT NULL,

JAIL\_ID INT(20) NOT NULL,

PRIMARY KEY (STID),

FOREIGN KEY (JAIL\_ID) REFERENCES PRISON(JAIL\_ID) ON DELETE CASCADE ON UPDATE CASCADE

);

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

---------------------------------------------**FOR TABLE SUPER\_USER**-----------------------------------------

CREATE TABLE SUPER\_USER

(

SUPERID INT(20) NOT NULL,

EMAIL\_ID VARCHAR(20) NOT NULL,

MOBILE\_NO NUMERIC(10) NOT NULL ,

SUPER\_NAME VARCHAR(20) NOT NULL,

WORK\_ID VARCHAR(20) NOT NULL,

JAIL\_ID VARCHAR(20) NOT NULL,

PASSWORD VARCHAR(20) NOT NULL,

PRIMARY KEY (SUPERID),

FOREIGN KEY (WORK\_ID) REFERENCES WORK\_ALLOTMENT(WORK\_ID) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (JAIL\_ID) REFERENCES PRISON(JAIL\_ID) ON DELETE CASCADE ON UPDATE CASCADE

);

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

----------------------------------------------**FOR TABLE PRISONER1**-------------------------------------------

CREATE TABLE PRISONER1

(

PRISONER\_ID INT(20) NOT NULL AUTO\_INCREMENT,

PNAME VARCHAR(20) NOT NULL,

DOB DATE NOT NULL,

RELEASE\_DATE DATE NOT NULL,

ADMISSION\_DATE DATE NOT NULL,

WORK\_ID INT(20) NOT NULL,

JAIL\_ID INT(20) NOT NULL,

CELL\_ID INT(20) NOT NULL,

CASETYPE VARCHAR(20) NOT NULL,

PRIMARY KEY (PRISONER\_ID),

FOREIGN KEY (WORK\_ID) REFERENCES WORK\_ALLOTMENT(WORK\_ID) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (JAIL\_ID) REFERENCES PRISON(JAIL\_ID) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (CELL\_ID) REFERENCES CELL(CELL\_ID)ON DELETE CASCADE ON UPDATE CASCADE

);

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

--------------------------------------------------**FOR TABLE MEDICAL**--------------------------------------------

CREATE TABLE MEDICAL

(

TREATMENT\_ID INT(20) NOT NULL AUTO\_INCREMENT,

ILLNESS VARCHAR(20) NOT NULL,

PRISONER\_ID VARCHAR(20) NOT NULL,

PRIMARY KEY (TREATMENT\_ID),

FOREIGN KEY (PRISONER\_ID) REFERENCES PRISONER1(PRISONER\_ID) ON DELETE CASCADE ON UPDATE CASCADE

);

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

-------------------------------------------------**FOR TABLE USER1**--------------------------------------------------

CREATE TABLE USER1

(

USERID INT(20) NOT NULL AUTO\_INCREMENT,

EMAIL VARCHAR(20),

FNAME VARCHAR(20) NOT NULL,

USERNAME VARCHAR(20) NOT NULL,

RELATION VARCHAR(20) ,

MOBILE NUMERIC(10) NOT NULL ,

GENDER VARCHAR(6) NOT NULL,

PASSWORD VARCHAR(20) NOT NULL,

ACCESS INT(20) NOT NULL,

PRISONER\_ID INT(20) NOT NULL,

PRIMARY KEY (USERID),

FOREIGN KEY (PRISONER\_ID) REFERENCES PRISONER1(PRISONER\_ID) ON DELETE CASCADE ON UPDATE CASCADE

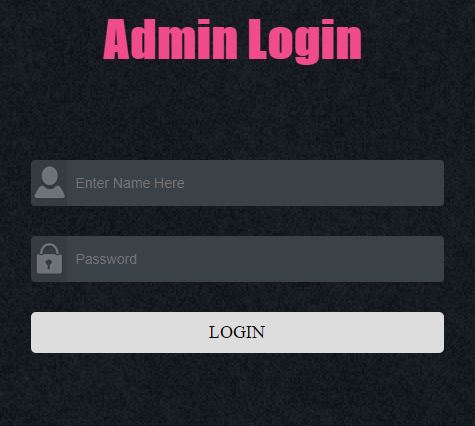
);

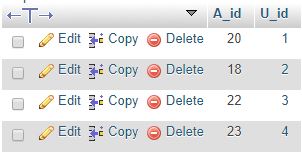
-----------------------------------------------------------------------------------------------------------------------------------

*INSERT QUERIES*

-----------------------------------------------**FOR TABLE AUTHORITY**------------------------------------------

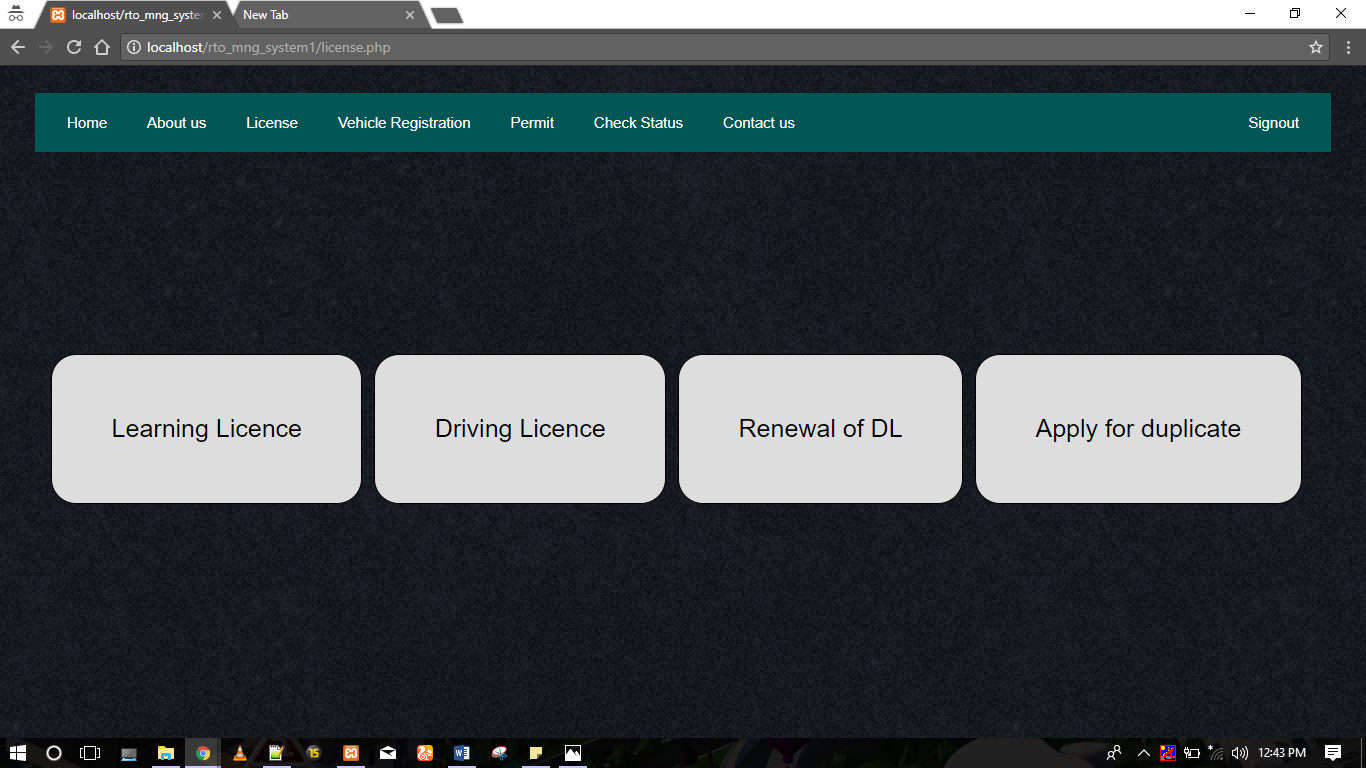
INSERT INTO `cell` (`CELL\_ID`, `JAIL\_ID`) VALUES (NULL, '1'), (NULL, '2'), (NULL, '3'), (NULL, '8'), (NULL, '5')

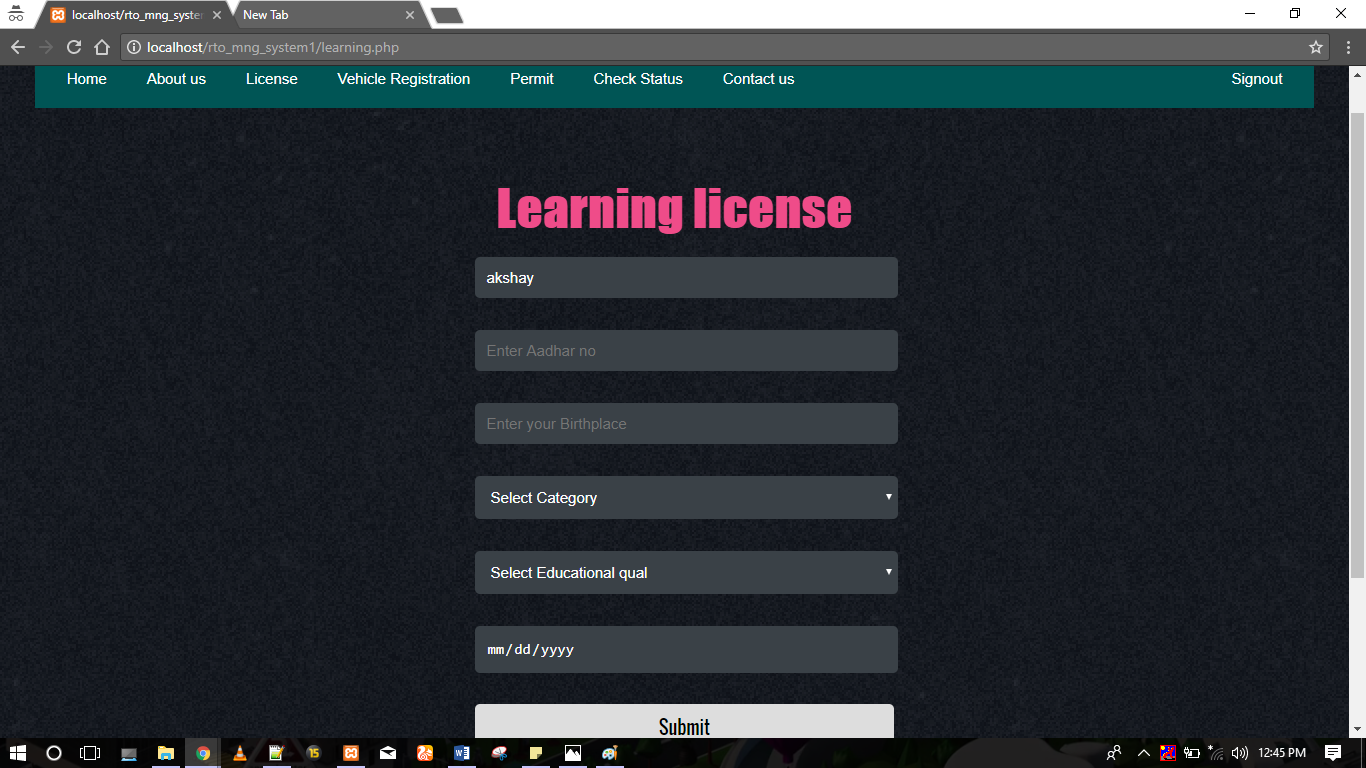




----------------------------------------------------------------------------------------------------------------------------------

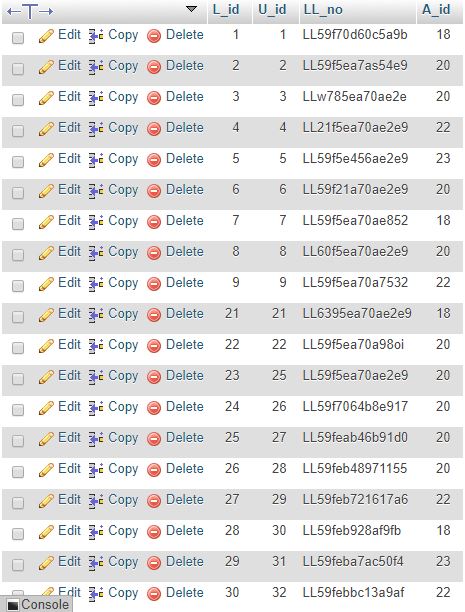
---------------------------------------**FOR TABLE LEARNING LISENCE**------------------------------------







------------------------------------------------**FOR TABLE LL\_RESULT**----------------------------------------



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

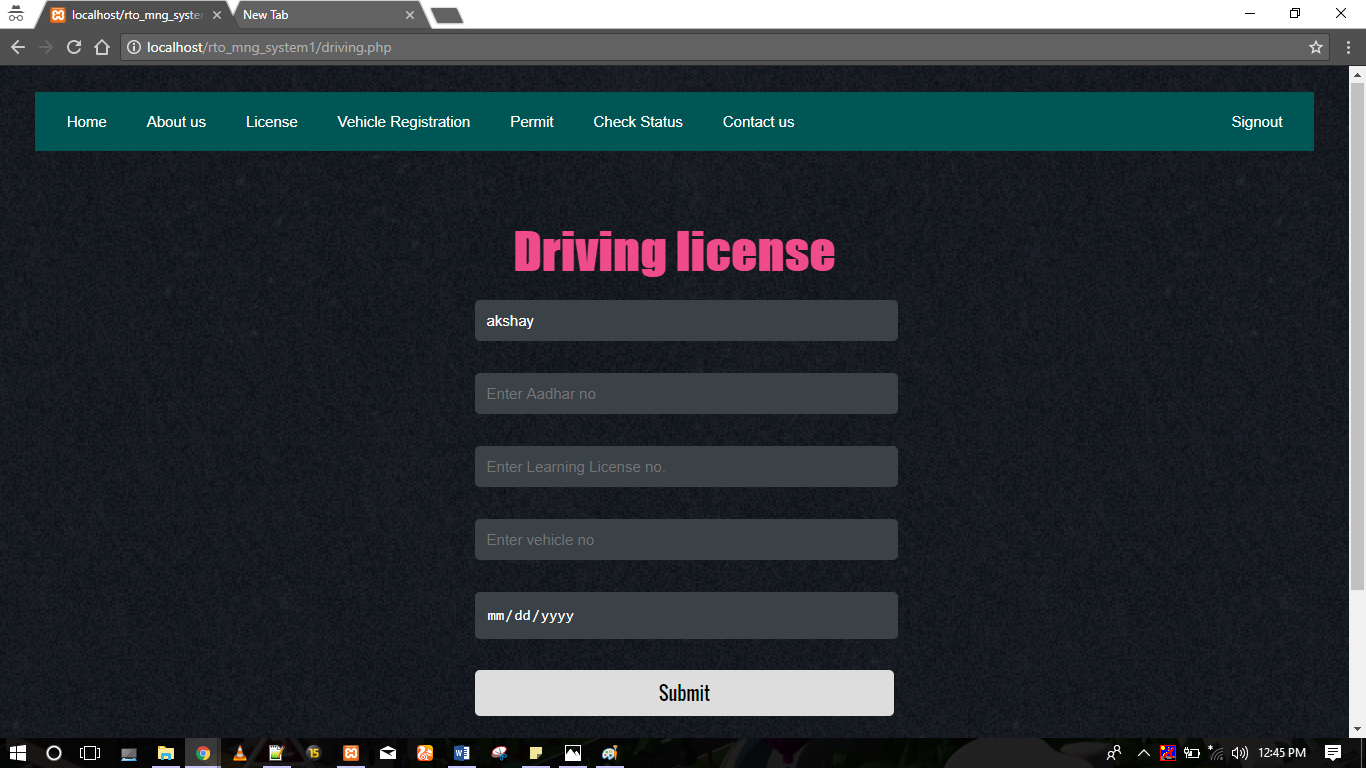
-------------------------------------------**FOR TABLE DRIVING LICENSE**------------------------------------

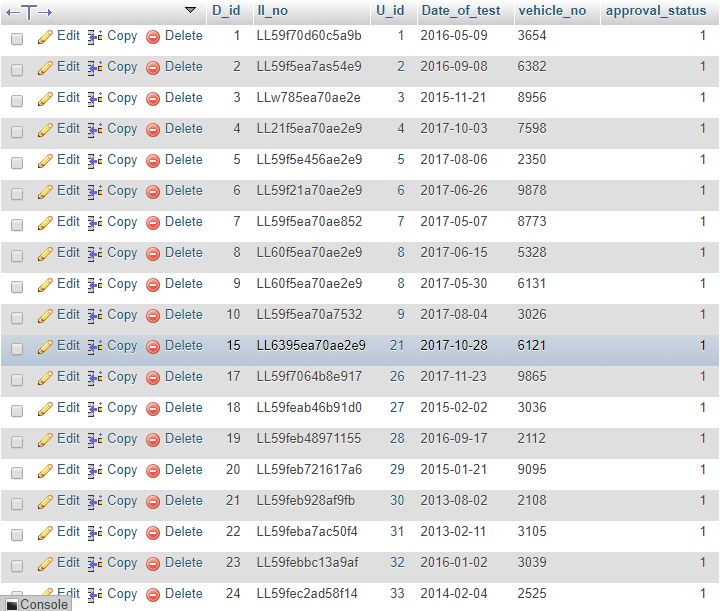
INSERT INTO `prison` (`PCAPACITY``, `JAIL\_ID`, `PTYPE`, `CONTACT\_NO`, `CCAPACITY`,`LOCATION`)

VALUES

('2', 'NULL', 'DISTRICT', '9898160747', ‘NULL’,’2’,’RAIYA ROAD,RAJKOT’),

('5', 'NULL', 'DISTRICT', '9638524047', NULL,’5’,’ADAJAN,SURAT)





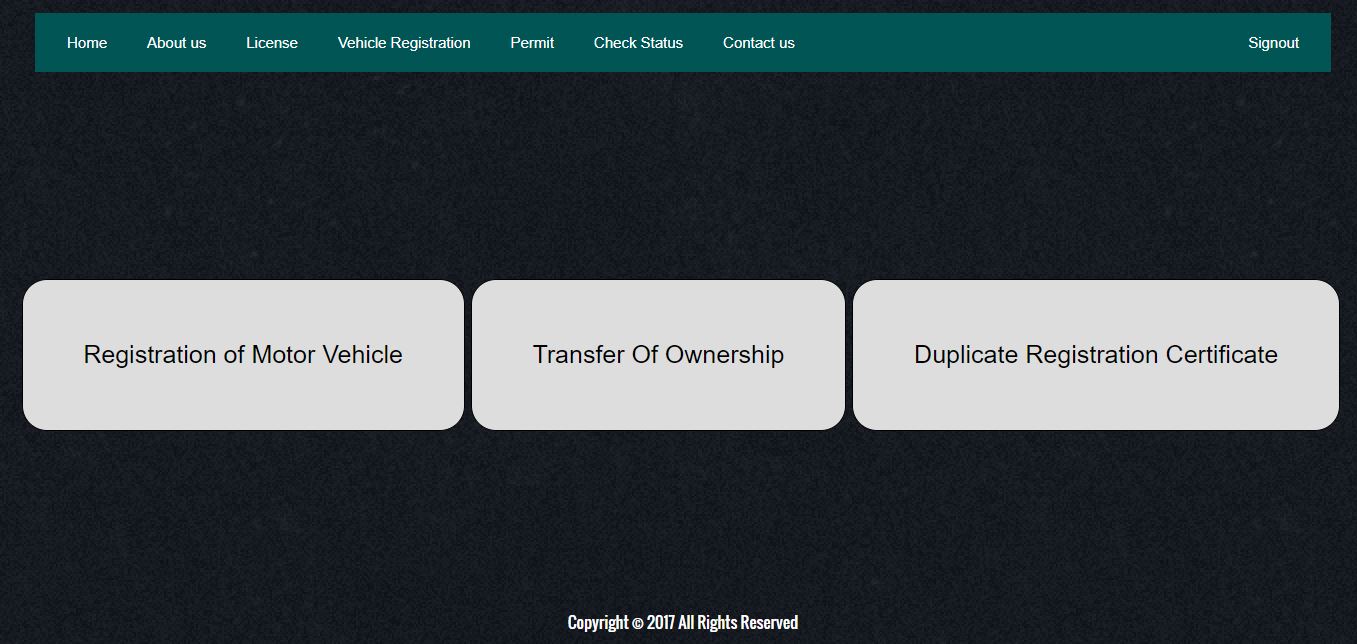
----------------------------------------------------------------------------------------------------------------------------------

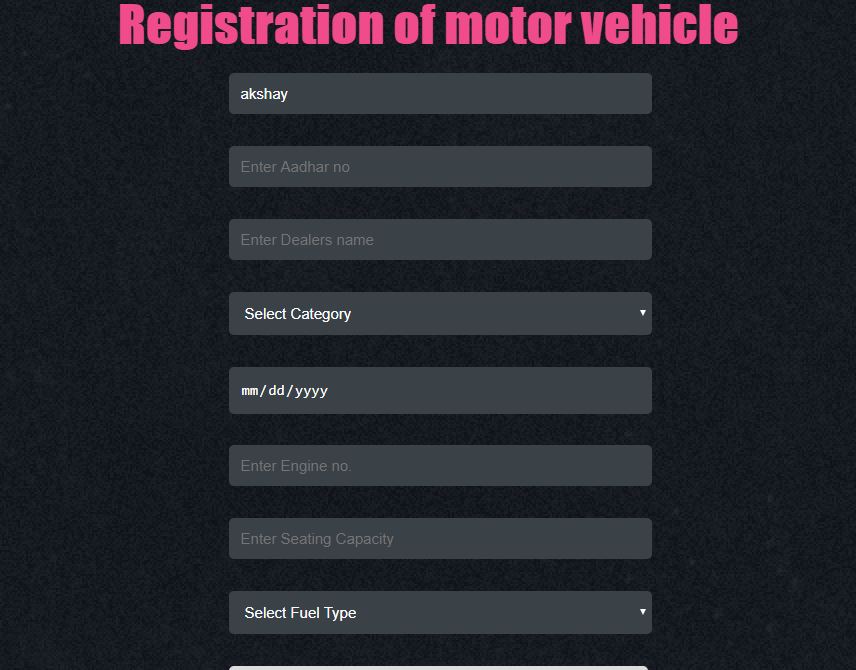
------------------------------------------------**FOR TABLE DL\_RESULT**-----------------------------------------



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

-------------------------**FOR TABLE REGISTRATION FOR MOTOR VEHICLE**-------------------







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

----------------------------**FOR TABLE REG\_MOTOR\_VEHICLE\_RESULT**--------------------------

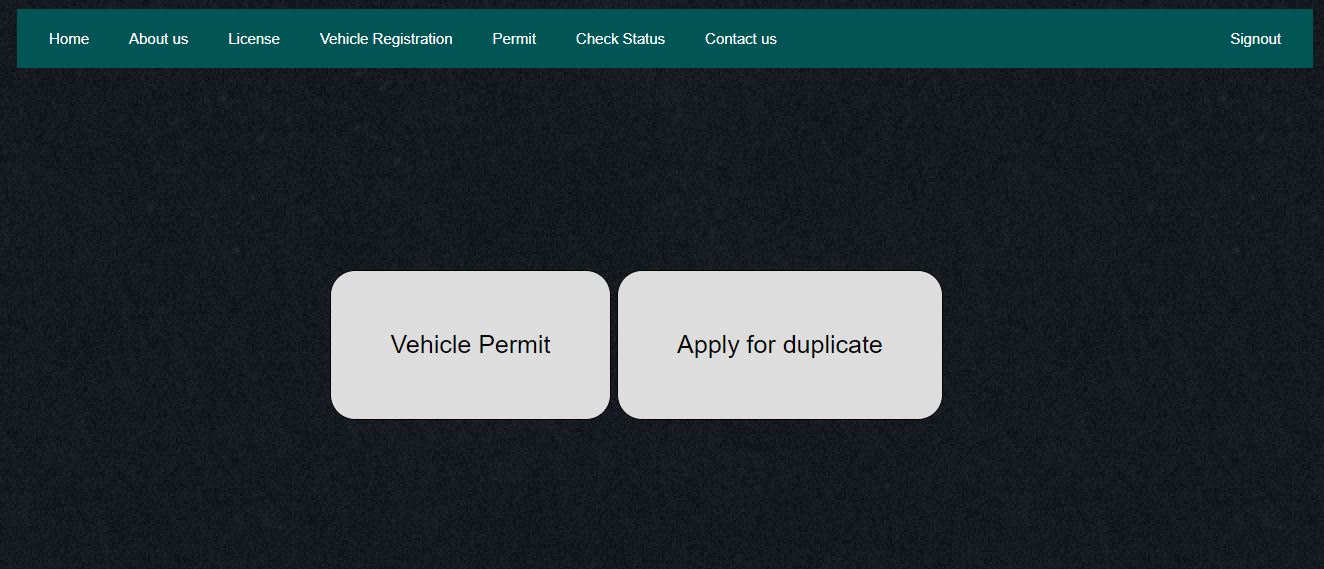
INSERT INTO `super\_user` (`SUPERID`, `EMAIL`, `MOBILE\_NO`, `SUPER\_NAME`, `WORK\_ID`, `JAIL\_ID`, `PASSWORD`) VALUES (NULL, 'milind@gmail.com', '741258963', 'milind', '1', '1', 'msp');

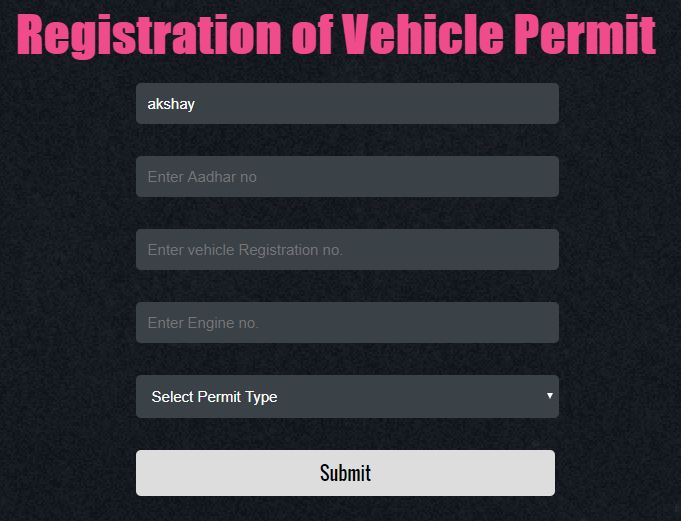
INSERT INTO `super\_user` (`SUPERID`, `EMAIL`, `MOBILE\_NO`, `SUPER\_NAME`, `WORK\_ID`, `JAIL\_ID`, `PASSWORD`) VALUES (NULL, 'nancy@gmail.com', '147852369', 'nancy', '3', '3', 'nsp');

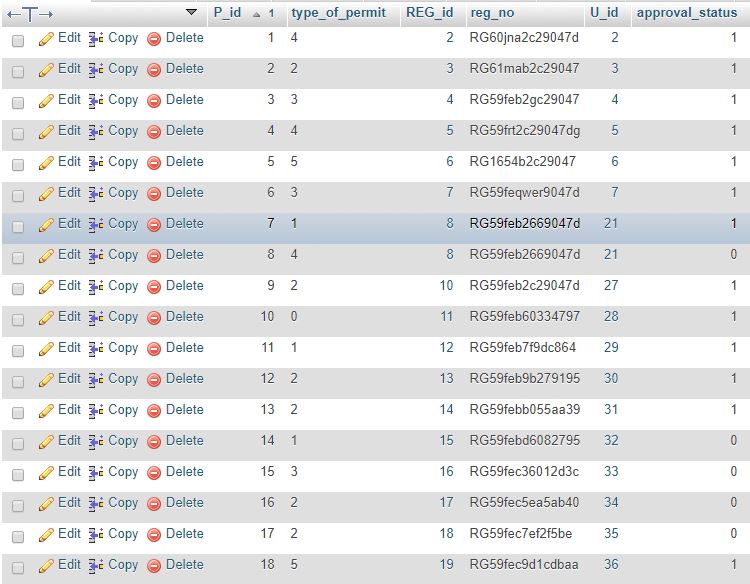
-----------------------------------------------------------------------------------------------------------------------------------

------------------------------------------------**FOR TABLE PERMIT**------------------------------------------------

Visitor can register through new registeration. The prisoner id of prisoner whom he/she wants to visit is available in view prisoner id link. Visitor can register successfully only if EMAIL,PASSWORD,MOBILE are entered properly.







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

---------------------------------------**FOR TABLE PERMIT\_RESULT**-----------------------------------------

INSERT INTO `work\_allotment` (`WTYPE`, `WAGES`, `WORK\_ID`)

VALUES ('CLEANING', '30', NULL),

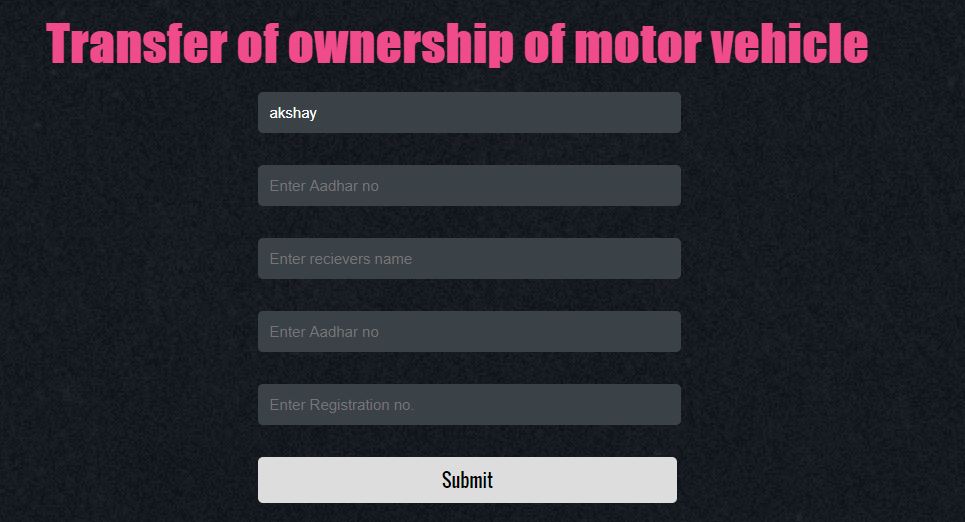
('LABOUR', '50', NULL),

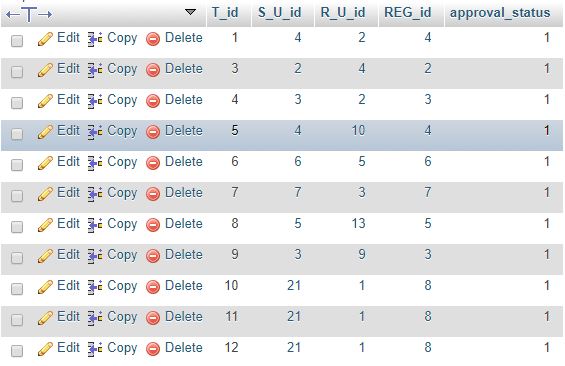
('FOOD', '30', NULL)



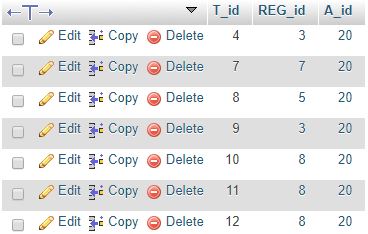
-----------------------------------------------------------------------------------------------------------------------------------

------------------------------**FOR TABLE TRANSFER OF OWNERSHIP**--------------------------------



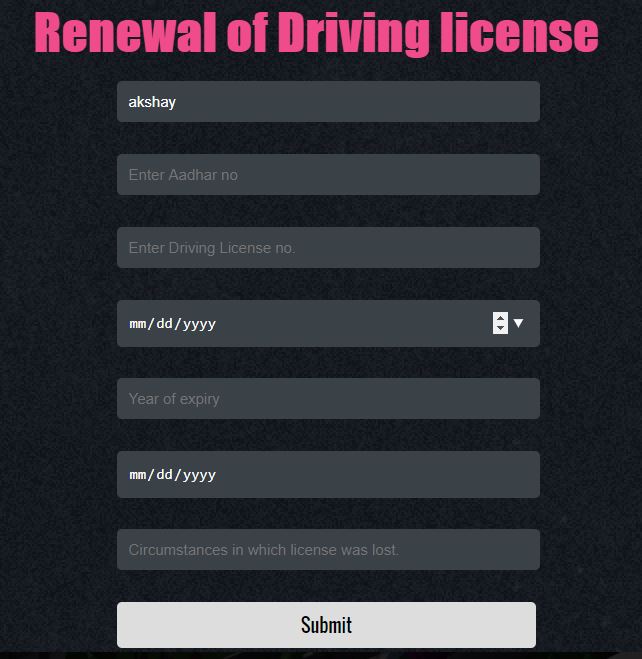
-----------------------------------------------------------------------------------------------------------------------------------

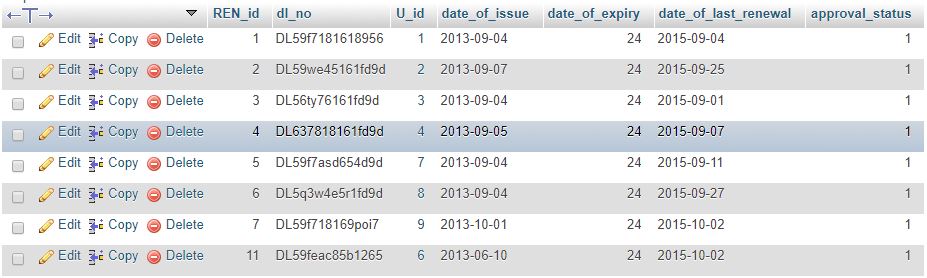
---------------------------------------**FOR TABLE TRANSFER PROCESS**---------------------------------



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

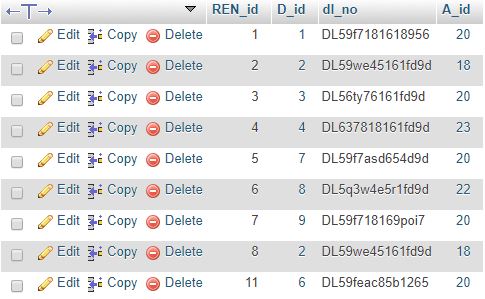
-----------------------------**FOR TABLE RENEWAL OF DRIVING LICENSE**-------------------------





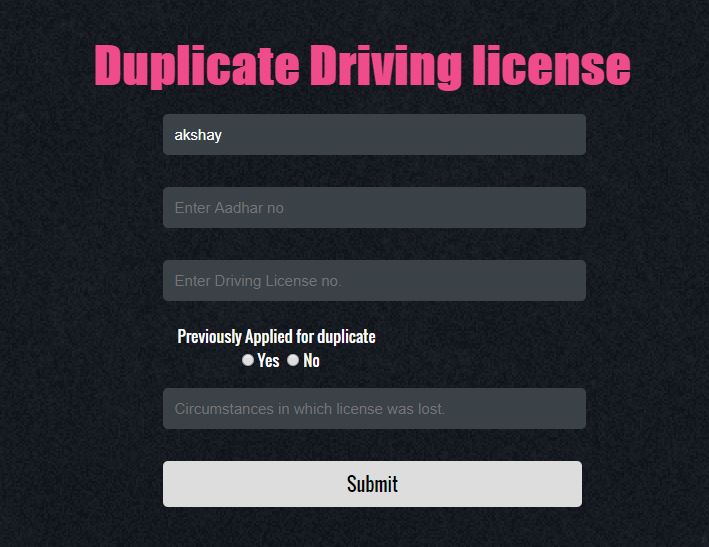
-----------------------------------------------------------------------------------------------------------------------------------

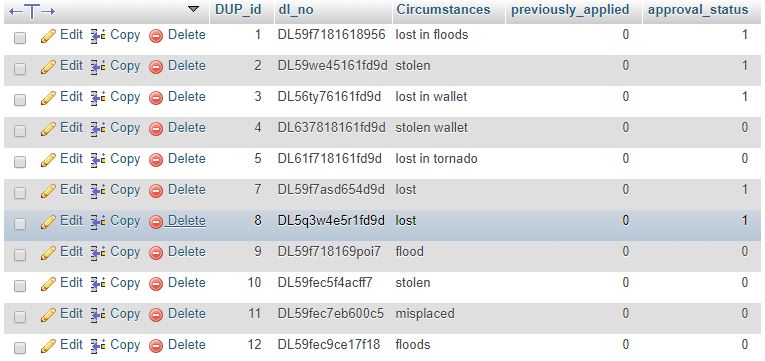
--------------------------------**FOR TABLE RENEWAL OF DL RESULT**---------------------------------



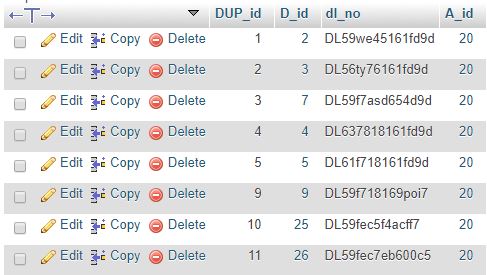
-----------------------------------------------------------------------------------------------------------------------------------

-------------------------------------**FOR TABLE DUPLICATE LICENSE**------------------------------------



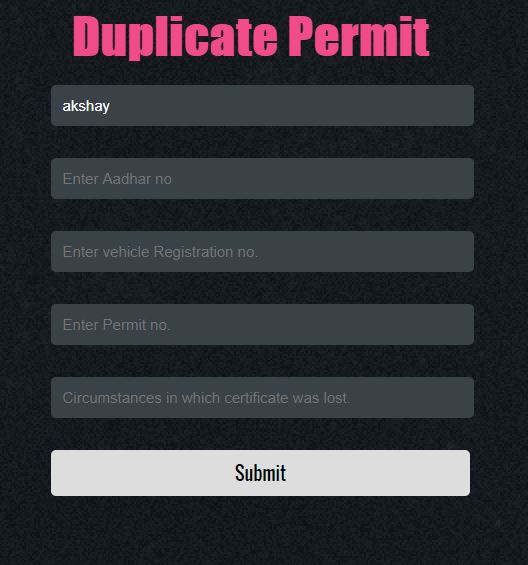
-----------------------------------------------------------------------------------------------------------------------------------

-------------------------------**FOR TABLE DUPLICATE LICENSE RESULT**---------------------------



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

--------------------------------------**FOR TABLE DUPLICATE PERMIT**-------------------------------------



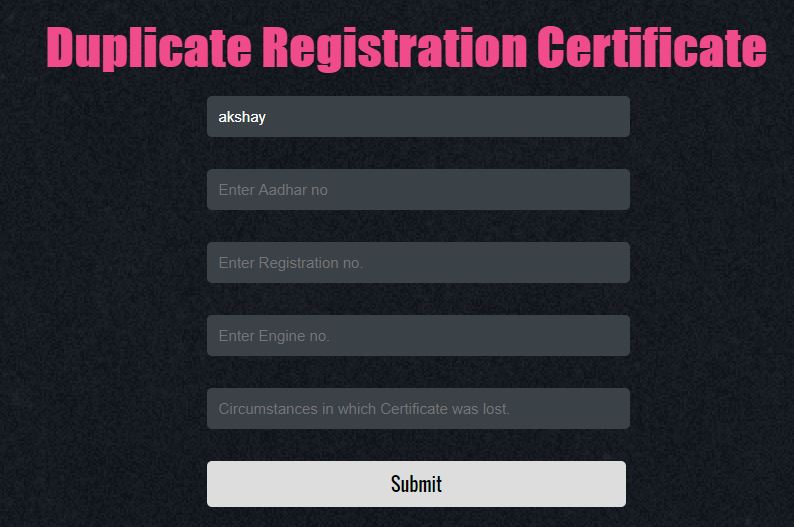
-----------------------------------------------------------------------------------------------------------------------------------

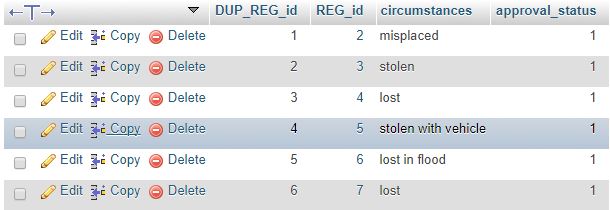
-------------------------------**FOR TABLE DUPLICATE PERMIT RESULT**-----------------------------



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

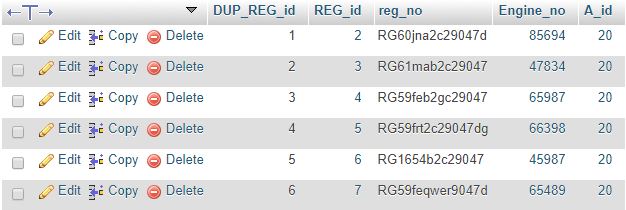
-----------------------------**FOR TABLE DUPLICATE REG CERTIFICATE**----------------------------





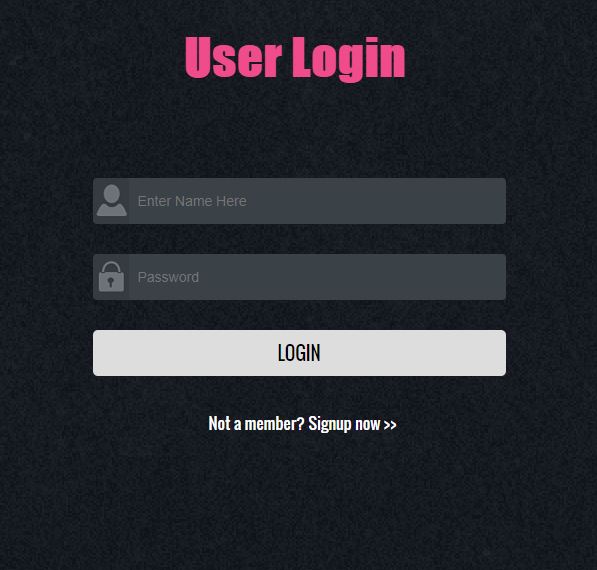
-----------------------------------------------------------------------------------------------------------------------------------

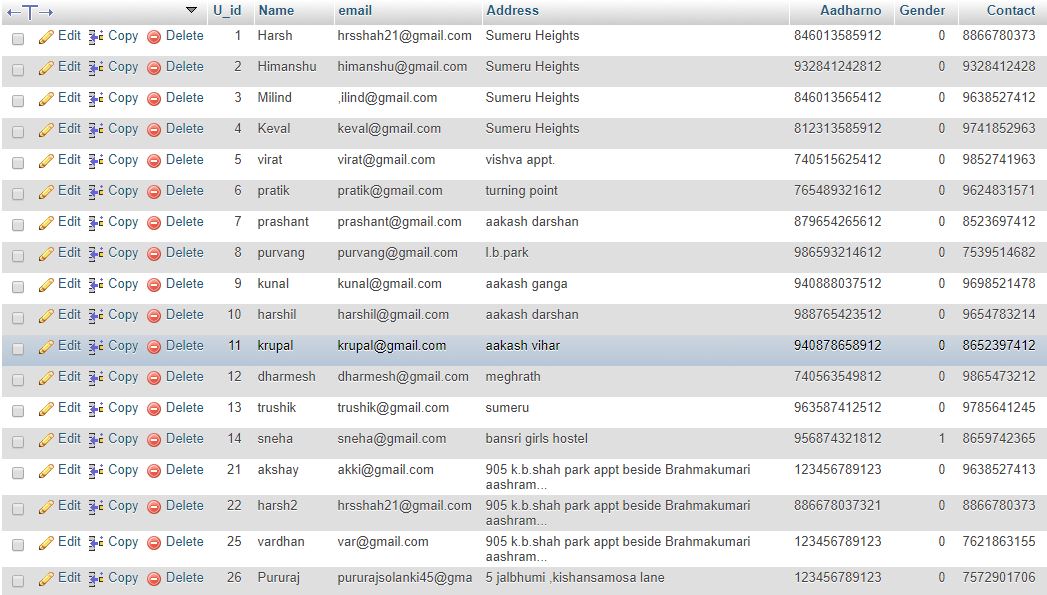
---------------------------------**FOR TABLE DUPLICATE REG RESULT**---------------------------------



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

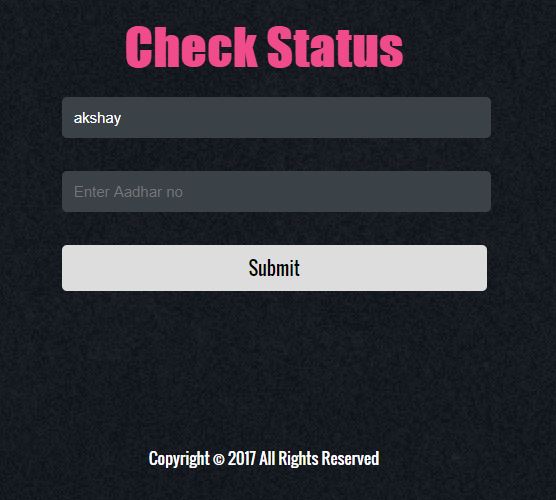
----------------------------------------------------**FOR TABLE USER**------------------------------------------------

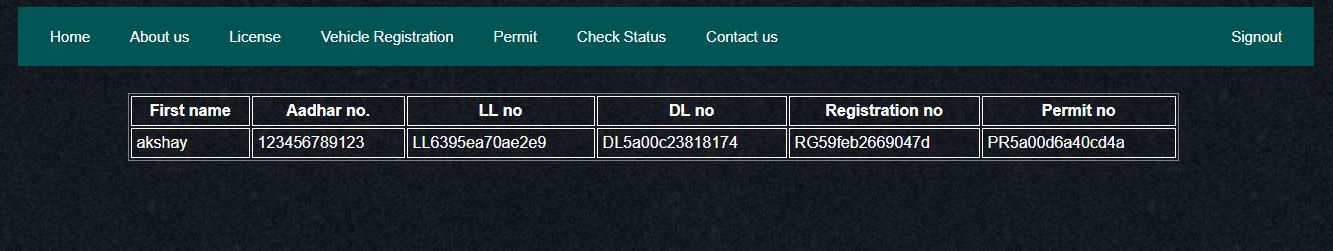




-----------------------------------------------------------------------------------------------------------------------------------

-------------------------------------------**FOR CHECKING STATUS**----------------------------------------------





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

*TRIGGERS*

--------------------**TRIGGER FOR ADMIN GRANTS PERMISSION TO VISITOR** -------------

When update button is clicked this trigger runs after update on permission table. Whenever a visitor requests to visit a prisoner his/her data entry is done in permission table. Now when admin grants the permission table by selecting ‘yes’ trigger is generated and access value which was updated in permission table is updated is updated in user table. Only when access in user table is set to ‘1’ ie admin grants the permission user will be able to see prisoner details.

PHP CODE USED FOR TRIGGER

<?php

if(isset($\_GET['Update']))

{

session\_write\_close();

$res1=mysqli\_query($conn,"select \* from permission WHERE ACCESS IS NULL LIMIT 1");

while($row=mysqli\_fetch\_array($res1))

{

$\_UserID=$row["UserID"];

session\_id("session2");

session\_start();

$\_SESSION["UserID"]=$\_UserID;

if(isset($\_GET['radio'])){

$gender=$\_GET['radio'];

}

**function ExecuteSetQuery($sql){**

**$con = mysqli\_connect ("localhost", "root","","test1"); // mysql\_select\_db ("mydb",$con);**

**$result = mysqli\_query($con, $sql) or die ("Query fail: ".mysqli\_error($con));**

**mysqli\_close($con);**

**return $result;**

**}**

**$sql = "DROP TRIGGER IF EXISTS `grant`;";**

**$res = ExecuteSetQuery($sql);**

**$sql = "CREATE TRIGGER `grant` AFTER UPDATE ON `permission`**

**FOR EACH ROW**

**BEGIN**

**IF EXISTS (SELECT \* FROM permission WHERE `UserID`=$\_SESSION[UserID] AND `ACCESS`=$\_GET[radio] LIMIT 1)**

**THEN UPDATE `user1` ACCESS SET ` `=$\_GET[radio] WHERE `UserID`=$\_SESSION[UserID];**

**END IF;**

**END";**

**$res = ExecuteSetQuery($sql);**

**$sql = $conn->query("UPDATE permission SET ACCESS = '{$gender}' where UserID=$\_SESSION[UserID]");**

**phpAlert( "You Have Successfully Granted a Permission" );**

**header("location: ". $\_SERVER['HTTP\_REFERER']);**

**}**

session\_write\_close();

session\_id("session1");

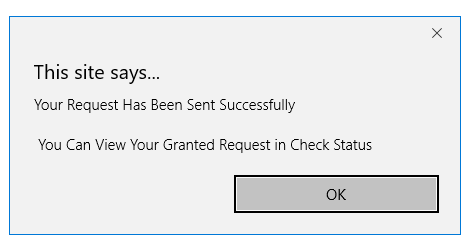
session\_start();

}

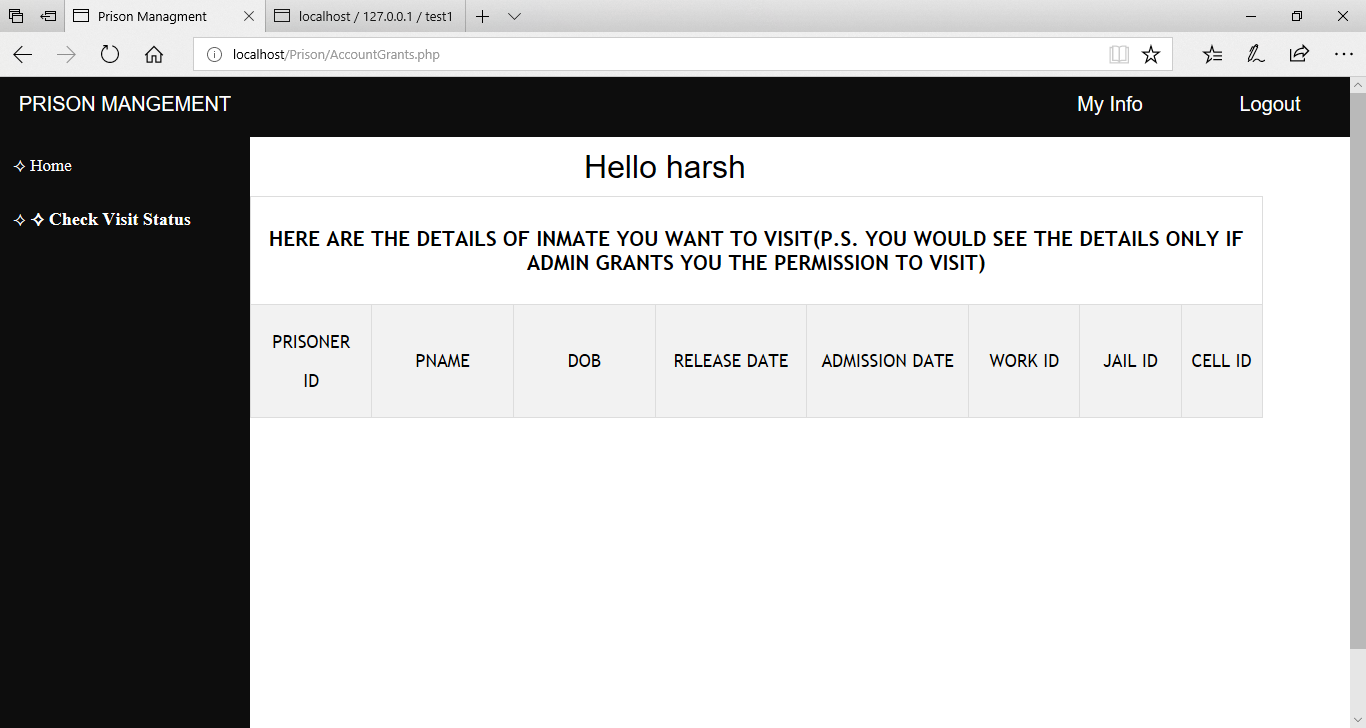
?>

Output:

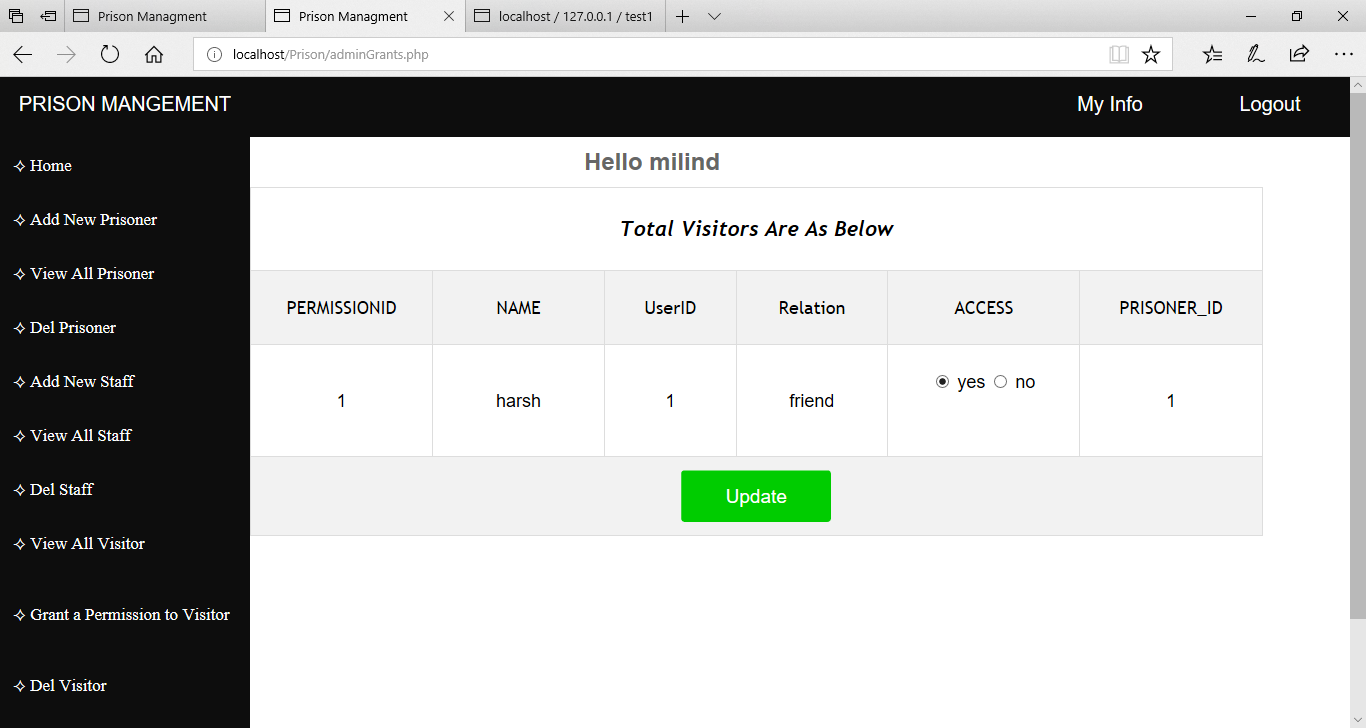
1. After requesting for visit

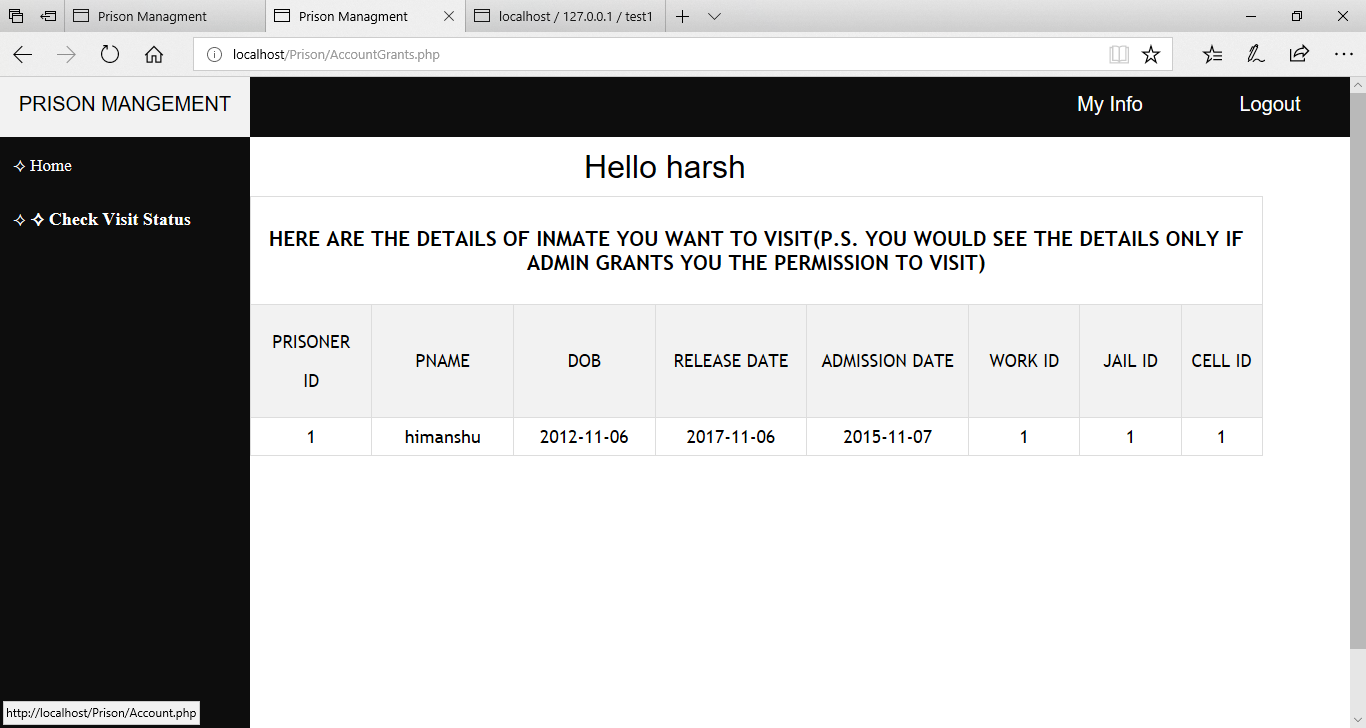


1. Before request granted



3.When admin grants permission



4. After admin grants permission

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

------**TRIGGER FOR DECREASE IN CURRENT CAPACITY OF PRISON WHEN**--------------------------------------------------- **PRISONER IS ADDED**-------------------------------------------------

After clicking on add prisoner button a trigger is generated before insert on prisoner1 table. A prisoner is added to the prison only if its current capacity is not zero ie no more space in prison. Whenever a prisoner is added the CCAPACITY of prison is decremented by 1 and min it can be 0. If CCAPACITY is 0 prisoner is not added to prison.

PHP CODE USED FOR TRIGGER

<?php

require 'connection.php';

require 'AlertMsg.php';

session\_start();

if(isset($\_SESSION["SUPERID"])){

$a=$\_SESSION["SUPERID"];

$sqlname = "SELECT SUPER\_NAME FROM super\_user where SUPERID=$a";

$resultname = mysqli\_query($conn, $sqlname);

if(isset($\_POST['submit'])){

$PNAME=$\_POST['PNAME'];

$GENDER=$\_POST['GENDER'];

$DOB=$\_POST['DOB'];

$ADMISSION\_DATE=$\_POST['ADMISSION\_DATE'];

$RELEASE\_DATE=$\_POST['RELEASE\_DATE'];

$WORK\_ID=$\_POST['WORK\_ID'];

$JAIL\_ID=$\_POST['JAIL\_ID'];

$CELL\_ID=$\_POST['CELL\_ID'];

$CASETYPE=$\_POST['CASETYPE'];

$res1=mysqli\_query($conn,"select CCAPACITY from prison WHERE JAIL\_ID=$JAIL\_ID");

while($row=mysqli\_fetch\_array($res1))

{

**$ABC=$row["CCAPACITY"];**

**if($ABC==0){**

**phpAlert( "You Can't Add a New Prisoner" );**

**}**

**else{**

**function ExecuteSetQuery($sql){**

**$con = mysqli\_connect ("localhost", "root","","test1");**

**$result = mysqli\_query($con, $sql) or die ("Query fail: ".mysqli\_error($con));**

**mysqli\_close($con);**

**return $result;**

**}**

**$sql = "DROP TRIGGER IF EXISTS `count`;";**

**$res = ExecuteSetQuery($sql);**

**$sql = "CREATE TRIGGER `count` BEFORE INSERT ON `prisoner1`**

**FOR EACH ROW**

**BEGIN**

**IF EXISTS (SELECT \* FROM prison WHERE `JAIL\_ID`=$\_POST[JAIL\_ID])**

**THEN**

**UPDATE prison SET CCAPACITY=CCAPACITY-1 WHERE `JAIL\_ID`=$\_POST[JAIL\_ID];**

**END IF;**

**END";**

**$res = ExecuteSetQuery($sql);**

if($ADMISSION\_DATE>$DOB and $RELEASE\_DATE>$DOB and $RELEASE\_DATE>$ADMISSION\_DATE){

$sql = $conn->query("INSERT INTO `prisoner1` (`PRISONER\_ID`, `PNAME`, `DOB`, `RELEASE\_DATE`, `ADMISSION\_DATE`, `WORK\_ID`, `JAIL\_ID`, `CELL\_ID`,`CASETYPE`)

Values(NULL,'{$PNAME}','{$DOB}','{$RELEASE\_DATE}','{$ADMISSION\_DATE}','{$WORK\_ID}','{$JAIL\_ID}','{$CELL\_ID}','{$CASETYPE}')");

phpAlert( "You Have Successfully Added a New Prisoner" );

}

else{

phpAlert( "Error !!\\n\\n Please Check The Dates You Have Entered!!" );

}

}}}

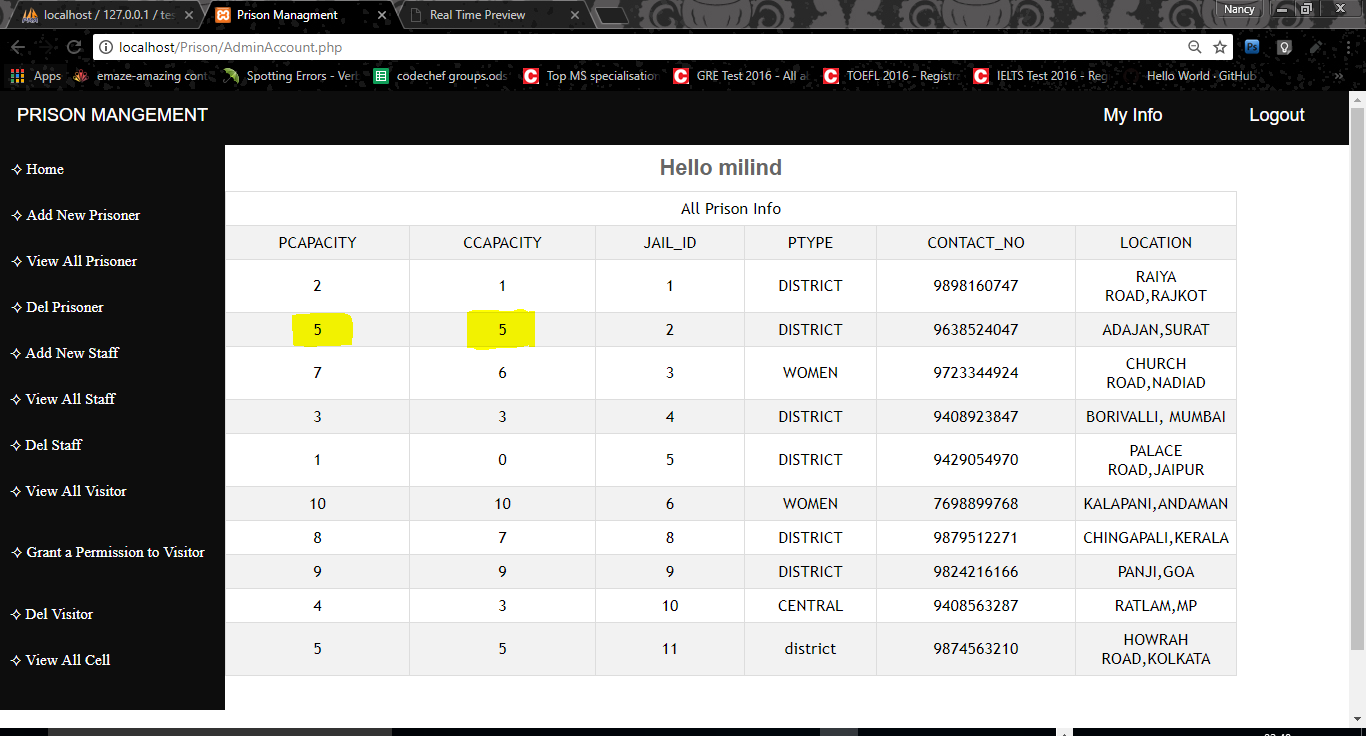
else{

header('location: AdminLogIn.php');

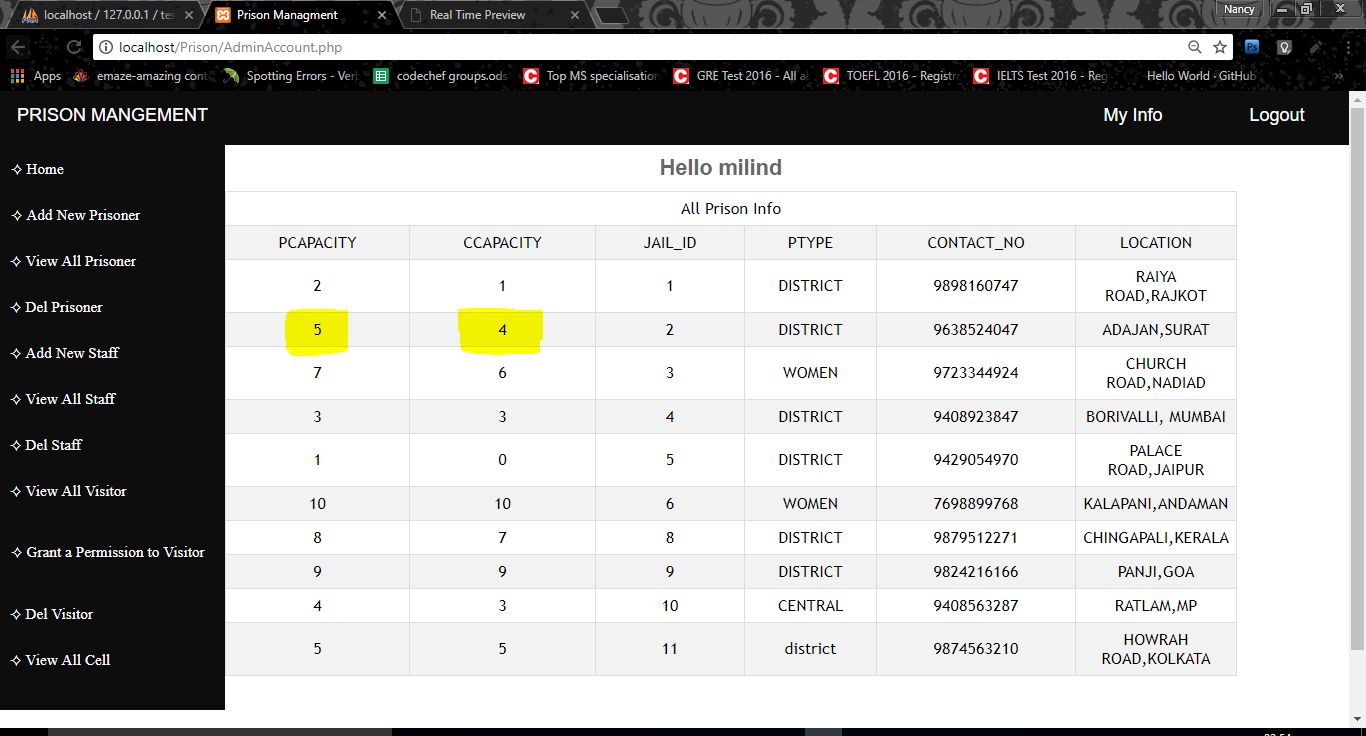
}

?>

Output:

Before adding prisoner

After adding prisoner



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

-----------------**TRIGGER FOR INCREASE IN CURRENT CAPACITY OF PRISON**-----------------------------------------------------------**WHEN A PRISONER IS RELEASED**----------------------------------------------

This trigger is generated on click on delete prisoner button after delete on prisoner1 table. Whenever a prisoner is released the current capacity of that prison is increased by 1. The current capacity of prison can not be increased more than the PCAPACITY of prison. A prisoner is released only when CCAPACITY is less or equal PCAPACITY.

PHP CODE USED FOR TRIGGER

<?php

require 'connection.php';

require 'AlertMsg.php';

session\_start();

if(isset($\_SESSION["SUPERID"])){

$a=$\_SESSION["SUPERID"];

$sqlname = "SELECT SUPER\_NAME FROM super\_user where SUPERID=$a";

$resultname = mysqli\_query($conn, $sqlname);

if(isset($\_POST['submit']))

{

$PRISONER\_ID=$\_POST['PRISONER\_ID'];

$res1=mysqli\_query($conn,"select JAIL\_ID from prisoner1 WHERE PRISONER\_ID=$PRISONER\_ID");

**while($row=mysqli\_fetch\_array($res1))**

**{**

**$JAIL\_ID=$row["JAIL\_ID"];**

**$res2=mysqli\_query($conn,"select CCAPACITY,PCAPACITY from prison WHERE JAIL\_ID=$JAIL\_ID");**

**while($row1=mysqli\_fetch\_array($res2))**

**{**

**$ABC=$row1["CCAPACITY"];**

**$qwe=$row1["PCAPACITY"];**

**if($ABC==$qwe){**

**}**

**else{**

**function ExecuteSetQuery($sql){**

**$con = mysqli\_connect ("localhost", "root","","test1");**

**$result = mysqli\_query($con, $sql) or die ("Query fail: ".mysqli\_error($con));**

**mysqli\_close($con);**

**return $result;**

**}**

**$sql = "DROP TRIGGER IF EXISTS `del`;";**

**$res = ExecuteSetQuery($sql);**

**$sql = "CREATE TRIGGER `del` AFTER DELETE ON `prisoner1`**

**FOR EACH ROW**

**BEGIN**

**UPDATE prison SET CCAPACITY=CCAPACITY+1 WHERE `JAIL\_ID`=$JAIL\_ID;**

**END";**

**$res = ExecuteSetQuery($sql);**

**$sql = $conn->query("DELETE FROM `prisoner1` WHERE `prisoner1`.`PRISONER\_ID` = '$PRISONER\_ID'");**

**phpAlert( "You Have Successfully Delete a Prisoner" );** }

}

}

}

}

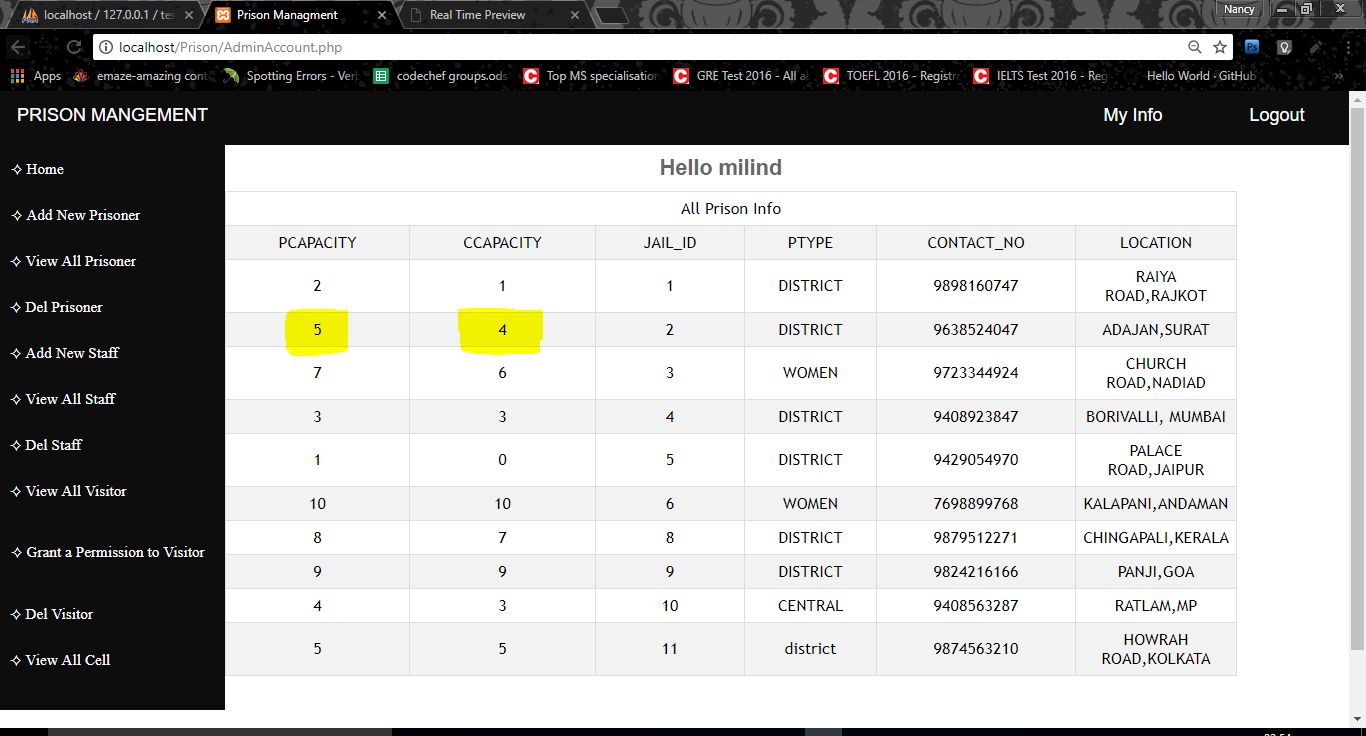
else{

header('location: AdminLogIn.php');

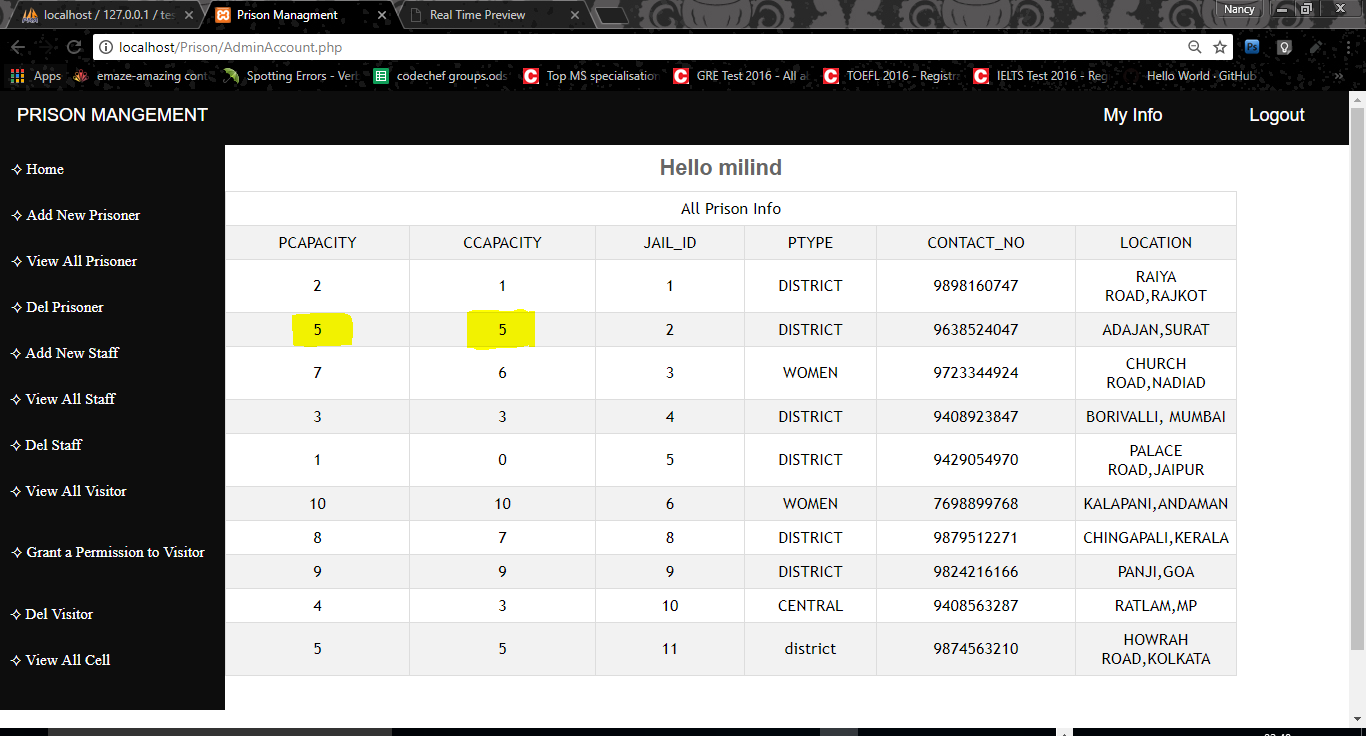
}

?>

Output:

Before deleting

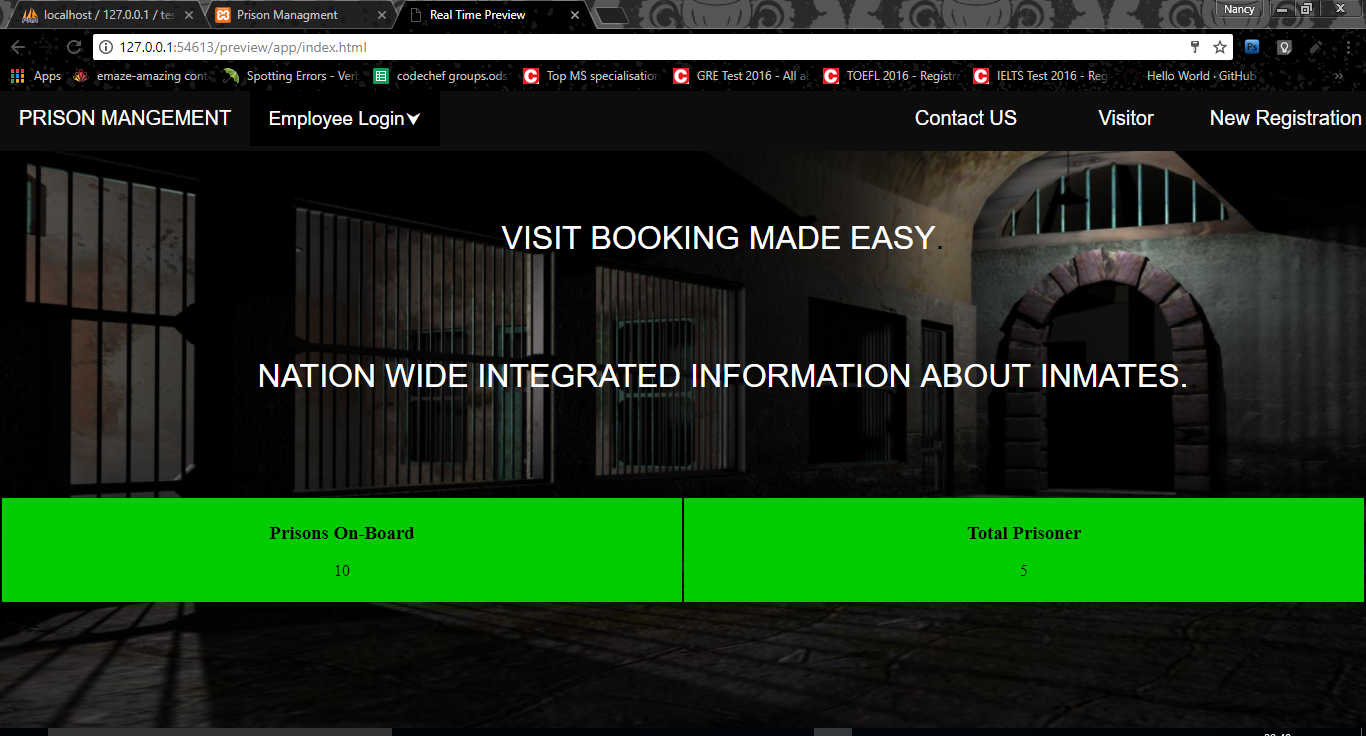
After delete



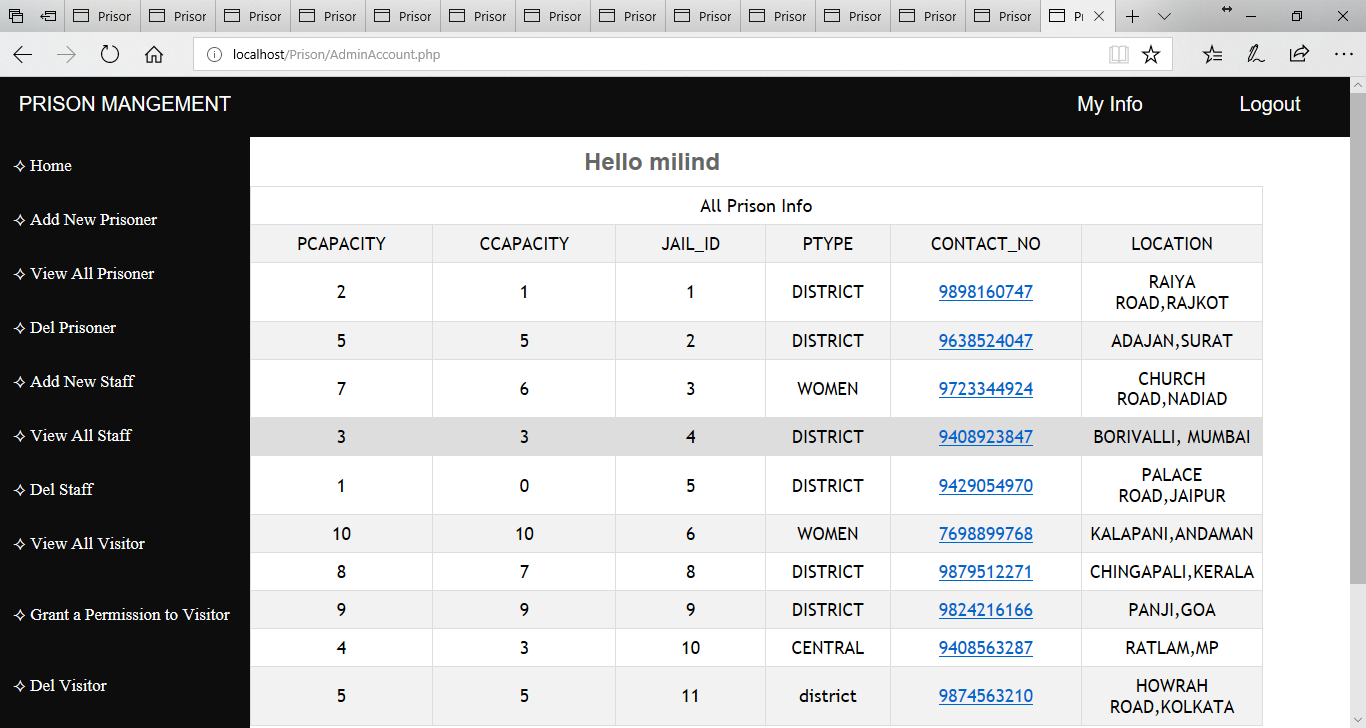
-----------------------------------------------------------------------------------------------------------------------------------

*OTHER FUNCTIONALITIES*

* Home page. The count query counts no. of rows in prison and prisoner table any displays it thus current status of prisons around the nation is known always



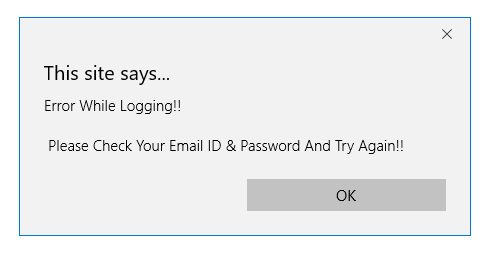
* When admin log’s in he/she able to see details of all prisons



* All visitor, admin and staff can update his/her info.



* If any data is wrongly entered while logging then such prompt arrives



* The work allotment of all prisoner ordered by work type is shown using subquery.



**BIBLIOGRAPHY**

* **Reference books:**
* **SQL, PL/SQL The programming language of Oracle**
  + Ivan Bayross, BPB Publications
* **Reference Link:**
* www.w3schools.com/
* <https://stackoverflow.com/>

* <https://www.w3resource.com/>