# PART III: IMPLEMENTATION

# 6 Table Creation Script

## 6.1 *<* WARDEN *>* TABLE

CREATE TABLE WARDEN (

Warden\_ID VARCHAR(30) NOT NULL PRIMARY KEY,

Fname varchar(15),

Lname varchar(15),

Gender varchar(20),

Address varchar(255),

Password NUMERIC(10),

Salary decimal(10,2) NOT NULL);

A screenshot of a computer

Description automatically generated with medium confidence

Not null constraint on salary since the Warden has to be paid.

## 6.2 *<* JOB\_SALARY *>* TABLE

CREATE TABLE JOB\_SALARY (

Job\_Type VARCHAR(15) NOT NULL PRIMARY KEY,

Salary decimal(10,2) NOT NULL);

Text

Description automatically generated

Not null constraint on salary since all jobs have a salary.

## 6.3 *<* STAFF *>* TABLE

CREATE TABLE STAFF(

staffID VARCHAR(30) NOT NULL PRIMARY KEY,

W\_ID VARCHAR(30) REFERENCES WARDEN (Warden\_ID) ,

Fname VARCHAR(20),

Lname VARCHAR(20),

Gender VARCHAR(20),

Job\_type VARCHAR(15) REFERENCES JOB\_SALARY(Job\_Type),

Address VARCHAR(255));

A screenshot of a computer

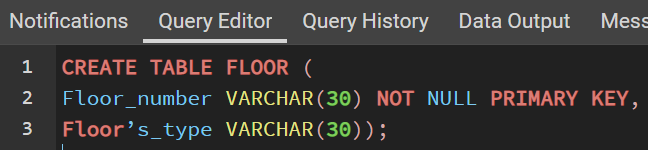
Description automatically generated with medium confidence

## 6.4 *<* FLOOR*>* TABLE

CREATE TABLE FLOOR (

Floor\_number VARCHAR(30) NOT NULL PRIMARY KEY,

Floor’s\_type VARCHAR(30));



## 6.5 *<* PRISON\_OFFICERS *>* TABLE

CREATE TABLE PRISON\_OFFICERS(

ID VARCHAR(30) NOT NULL PRIMARY KEY,

F\_number VARCHAR(30) REFERENCES FLOOR (Floor\_number),

FOREIGN KEY (ID) REFERENCES STAFF (staffID) ON DELETE CASCADE);

Text

Description automatically generated

## 6.6 *<* PRISON\_COOK *>* TABLE

CREATE TABLE PRISON\_COOK (

ID VARCHAR(30) NOT NULL PRIMARY KEY,

FOREIGN KEY (ID) REFERENCES STAFF (staffID) ON DELETE CASCADE);

Text

Description automatically generated

## 6.7 *<* HAED\_DOCTOR *>* TABLE

CREATE TABLE HEAD\_DOCTOR(

ID VARCHAR(30) NOT NULL PRIMARY KEY,

FOREIGN KEY (ID) REFERENCES STAFF (staffID) ON DELETE CASCADE);

Text

Description automatically generated

## 6.8 *<* YARD\_OFFICER *>* TABLE

CREATE TABLE YARD\_OFFICERS (

ID VARCHAR(30) NOT NULL PRIMARY KEY,

FOREIGN KEY (ID) REFERENCES STAFF (staffID) ON DELETE CASCADE);

Text

Description automatically generated

## 6.9 *<* CELLS *>* TABLE

CREATE TABLE CELLS(

Cells\_number NUMERIC(10) NOT NULL PRIMARY KEY,

F\_number VARCHAR(30) REFERENCES FLOOR(floor\_number),

Cells\_Capacity NUMERIC(10) DEFAULT 2,

Cells\_state varchar(20));

Text

Description automatically generatedSince all cell capacities are the same, DEFAULT is used.

Default ‘unoccupied’ is used, until a prisoner occupies the cell.

## 6.10 *<* PRISONERS *>* TABLE

CREATE TABLE PRISONERS(

Prisoner\_number NUMERIC(10) NOT NULL PRIMARY KEY,

Fname varchar(20),

Lname varchar(20),

Date\_of\_Arrival Date,

Date\_of\_Release Date,

C\_number numeric(10),

FOREIGN KEY (C\_number) REFERENCES CELLS (Cells\_number) ON DELETE CASCADE);

Text

Description automatically generated

## 6.11 *<* HEALTH\_RECORDS *>* TABLE

CREATE TABLE HEALTH\_RECORDS(

Prisoner\_name varchar(50) NOT NULL,

P\_number NUMERIC(10) NOT NULL,

State varchar(20),

Blood\_type varchar(20),

Date\_of\_birth date,

Weight decimal(10,2),

Height decimal(10,2),

Disease\_history varchar(255),

PRIMARY KEY(Prisoner\_name, P\_number),

FOREIGN KEY (P\_number) REFERENCES PRISONERS (Prisoner\_number) ON DELETE CASCADE);

A screenshot of a computer

Description automatically generated with medium confidence

(Prisoner\_name) is partial key and (p\_number) is a foreign key; those keys together will form the primary key for the weak entity.

## 6.12 *<* CRIMINAL\_RECORDS *>* TABLE

CREATE TABLE CRIMINAL\_RECORDS(

Prisoner\_name varchar(50) NOT NULL ,

P\_number NUMERIC(10) NOT NULL,

Prisoners\_crime varchar(50) NOT NULL,

PRIMARY KEY(Prisoner\_name, P\_number),

FOREIGN KEY (P\_number) REFERENCES PRISONERS (Prisoner\_number) ON DELETE CASCADE);

Text

Description automatically generated

(Prisoner\_name) is partial key and (p\_number) is a foreign key; those keys together will form the primary key for the weak entity.

Since all prisoners have committed crimes to enter the jail, prisoners\_crime must be NOT NULL.

## 6.13 *<* VISITS *>* TABLE

CREATE TABLE VISITS(

Visitor\_ID VARCHAR(30) NOT NULL,

P\_number NUMERIC(10),

Visitor\_name varchar(50),

Visitor\_phone\_number NUMERIC(15),

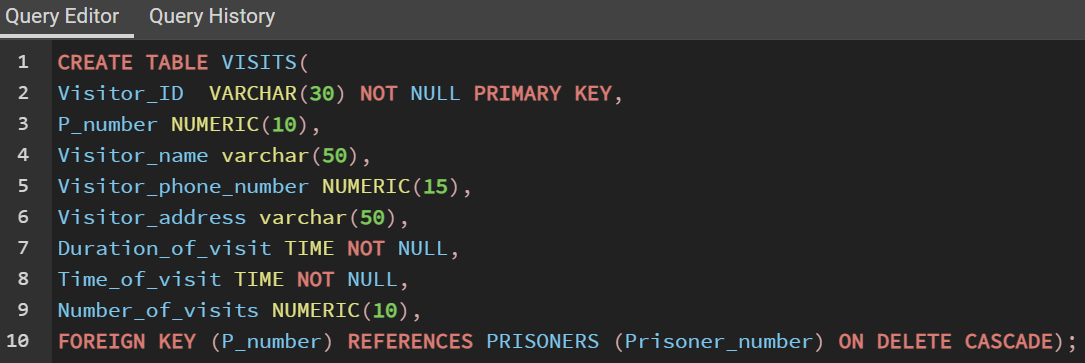
Visitor\_address varchar(50),

Duration\_of\_visit TIME NOT NULL,

Time\_of\_visit TIME NOT NULL,

Number\_of\_visits NUMERIC(10),

FOREIGN KEY (P\_number) REFERENCES PRISONERS (Prisoner\_number) ON DELETE CASCADE);



Since all visitors are allowed to visit prisoners for a limited time, Time\_of\_visit and Duration\_of\_visit must be measured to not break the prison rules, therefore the two attributes will have the constraint NOT NULL.

## 6.14 *<* ACTIVITY\_REWARD *>* TABLE

CREATE TABLE ACTIVITY\_REWARD(

Activity\_Type varchar(50) NOT NULL PRIMARY KEY,

Reward varchar(50) );

Text

Description automatically generated

## 6.15 *<* ACTIVITIES *>* TABLE

CREATE TABLE ACTIVITIES(

Activity\_number NUMERIC(10) NOT NULL PRIMARY KEY,

Activity\_Type varchar(50) REFERENCES ACTIVITY\_REWARD (Activity\_Type),

P\_number NUMERIC(10) REFERENCES PRISONERS (Prisoner\_number) );

Text

Description automatically generated

## 6.16 *<* MANAGED *>* TABLE

CREATE TABLE MANAGED (

P\_number NUMERIC(10) NOT NULL,

S\_ID varchar(30) NOT NULL,

PRIMARY KEY(P\_number, S\_ID) ,

FOREIGN KEY (P\_number) REFERENCES PRISONERS (Prisoner\_number)ON DELETE CASCADE,

FOREIGN KEY (S\_ID) REFERENCES STAFF (staffID) ON DELETE CASCADE );

Text

Description automatically generated

## 6.17 *<* CELLS\_FLOOR *>* TABLE

CREATE TABLE CELLS\_FLOOR(

F\_number VARCHAR(30) NOT NULL PRIMARY KEY,

NumberOFcells NUMERIC(100),

FOREIGN KEY (F\_number) REFERENCES FLOOR (Floor\_number) ON DELETE CASCADE );

Text

Description automatically generated

## 6.18 *<* WARDEN\_FLOOR *>* TABLE

CREATE TABLE WARDEN\_FLOOR (

F\_number VARCHAR(30) NOT NULL PRIMARY KEY,

Office\_number NUMERIC(10),

FOREIGN KEY (F\_number) REFERENCES FLOOR (Floor\_number) ON DELETE CASCADE );

Text

Description automatically generated

## 6.19 *<* PUNISHMENT *>* TABLE

CREATE TABLE PUNISHMENT (

Punishment\_number NUMERIC(10) NOT NULL PRIMARY KEY,

W\_ID VARCHAR(30) DEFAULT 200000,

Punishment\_type varchar(255),

FOREIGN KEY (W\_ID) REFERENCES Warden(Warden\_ID) ON DELETE CASCADE);

Text

Description automatically generated

Since (W\_ID) is the always the same, DEFAULT is used.

## 6.20 *<* LAW\_VIOLATORS*>* TABLE

CREATE TABLE LAW\_VIOLATERS (

Prisoners\_name varchar(50) NOT NULL ,

Punishment\_number NUMERIC(10) NOT NULL ,

PRIMARY KEY(Prisoners\_name, Punishment\_number),

FOREIGN KEY (Punishment\_number) REFERENCES PUNISHMENT(Punishment\_number) ON DELETE CASCADE);

Text

Description automatically generated

(Prisoners\_name) is partial key and (punishment\_number) is a foreign key; those keys together will form the primary key for the weak entity.

# 7 Constraints Script

This section will show some of the constraints we've put in place in our database. Keep in mind that these aren't the only database restrictions. It is merely a representation of some of the most fundamental and significant ones.

|  |  |  |
| --- | --- | --- |
| Business Rule | SQL Script | Table |
| Each staff member has a unique ID | Implementation: while creating the STAFF table  staffID VARCHAR(30) **NOT NULL PRIMARY KEY** | STAFF |
| Subclasses of STAFF must have an inherited ID. Since they are specializations of the superclass STAFF, if a STAFF member is deleted the data of all the referencing relations are deleted as well | FOREIGN KEY (ID) REFERENCES STAFF (staffID) **ON DELETE CASCADE**); | PRISON\_OFFICERS  YARD\_OFFICERS  HEAD\_DOCTOR  PRISON\_COOK |
| Each CRIMINAL RECORD has the PRISONER’S crime | Prisoners\_crime varchar(50) **NOT NULL** | CRIMINAL\_RECORDS |
| Each Cell that has fewer prisoners in them than the cell’s capacity is given a default label of “unoccupied” | Cells\_state varchar(20) **DEFAULT 'Unoccupied'**); | CELLS |

# 8 Queries and Transactions

The majority of our important queries and transactions do not need to be complicated. However, the system may require some more complicated queries and transactions.

## 8.1 <Average salaries >

**Query in Natural Language (English)**

This query will retrieve the average salary of each job type with an average salary greater than the specified department average salary.

Our database will be used in this context by users to get the average salary of each essential job type for the STAFF members (Prison officers, yard officers, head doctor, prison cook).

**SQL Script**

SELECT AVG(salary) AS "Ave Salary", job\_type

FROM job\_salary

Group by job\_type

HAVING AVG(salary) > (select avg(salary)

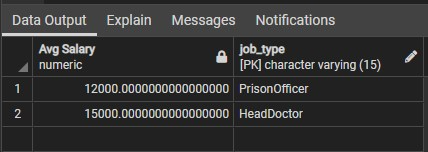
FROM job\_salary

Where job\_type='YardOfficer');

Text

Description automatically generated

**Caption of the First Five Rows of the Output**



## 8.2 *<Specify prisoners with certain characteristics>*

**Query in Natural Language (English)**

This query will display the Prisoner number of all prisoners who will be released after the date

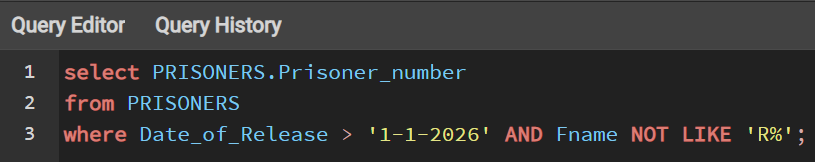
(1-1-2026), and their initials are not the letter (R).

**SQL Script**

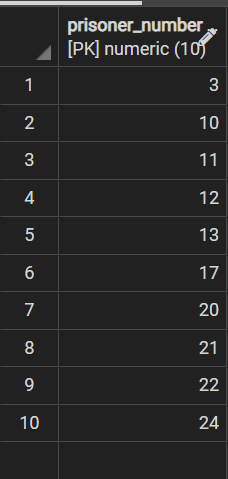
select PRISONERS.Prisoner\_number

from PRISONERS

where Date\_of\_Release > '1-1-2026' AND Fname NOT LIKE 'R%';



**Caption of the First Five Rows of the Output**



## 8.3 *<Specify robbers>*

**Query in Natural Language (English)**

Print ID, first name, and last name of prisoners who have committed robbery.

**SQL Script**

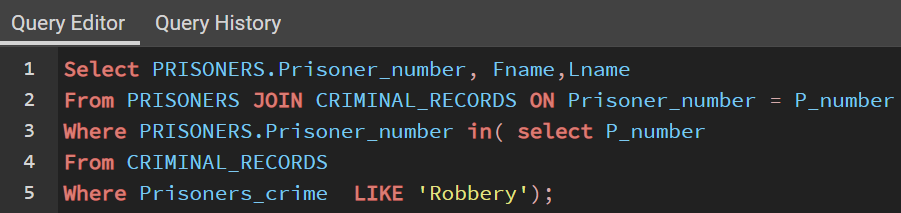
Select PRISONERS.Prisoner\_number, Fname,Lname

From PRISONERS JOIN CRIMINAL\_RECORDS ON Prisoner\_number = P\_number

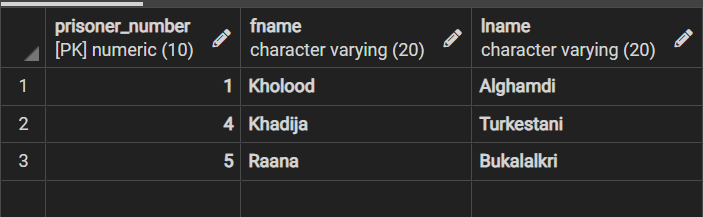
Where PRISONERS.Prisoner\_number in( select P\_number

From CRIMINAL\_RECORDS

Where Prisoners\_crime LIKE 'Robbery');



**Caption of the First Five Rows of the Output**



## 8.4 *<Salary Raise>*

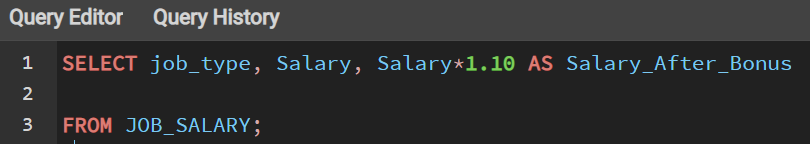
**Query in Natural Language (English)**

This query will raise the salary of all employees by 10%.

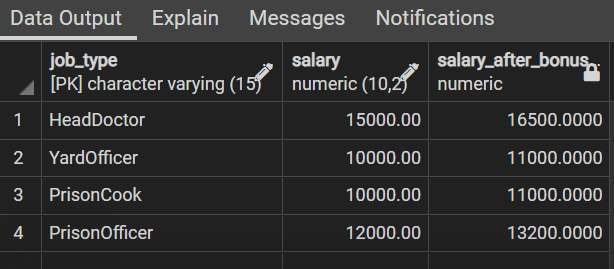
**SQL Script**

SELECT job\_type, Salary, Salary\*1.10 AS Salary\_After\_Bonus

FROM JOB\_SALARY;



**Caption of the First Five Rows of the Output**



## 8.5 *<Specify jobs with less than two members >*

**Query in Natural Language (English)**

This query will display jobs that have less than 2 members working in them including their names.

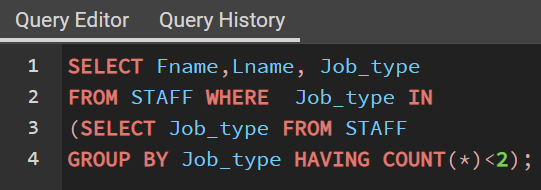
**SQL Script**

SELECT Fname,Lname, Job\_type

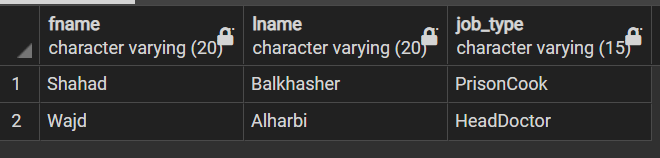
FROM STAFF WHERE Job\_type IN

(SELECT Job\_type FROM STAFF

GROUP BY Job\_type HAVING COUNT(\*)<2);



**Caption of the First Five Rows of the Output**



## 8.6*<Total Number of Staff Members Per Job Type>*

**Query in Natural Language (English)**

This query will display names of the jobs and the numbers of staff memebrs working in descending order. We used the aggregate function COUNT here to find the total number of members working in each job type.

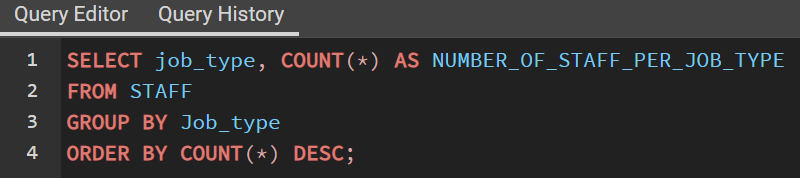
**SQL Script**

SELECT job\_type, COUNT(\*) AS NUMBER\_OF\_STAFF\_PER\_JOB\_TYPE

FROM STAFF

GROUP BY Job\_type

ORDER BY COUNT(\*) DESC;



**Caption of the First Five Rows of the Output**

Table

Description automatically generated

## 8.7*<The Largest Number of Staff Members in a Job Type>*

**Query in Natural Language (English)**

This query will display the largest number of staff members that are working in a job among all job types. We used nested aggregate functions (MAX, COUNT).

**SQL Script**

SELECT MAX(NUMBER\_OF\_STAFF\_PER\_JOB\_TYPE) AS "The Largest Number of Staff Members in a Job Type"

FROM(SELECT Job\_type, COUNT(\*)AS NUMBER\_OF\_STAFF\_PER\_JOB\_TYPE

FROM STAFF

GROUP BY Job\_type)

AS SubQueryAlias;

Text

Description automatically generated

**Caption of the First Five Rows of the Output**

Graphical user interface, text, application

Description automatically generated

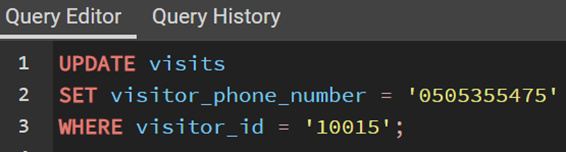
## 8.8 Update Example

### 8.8.1 example

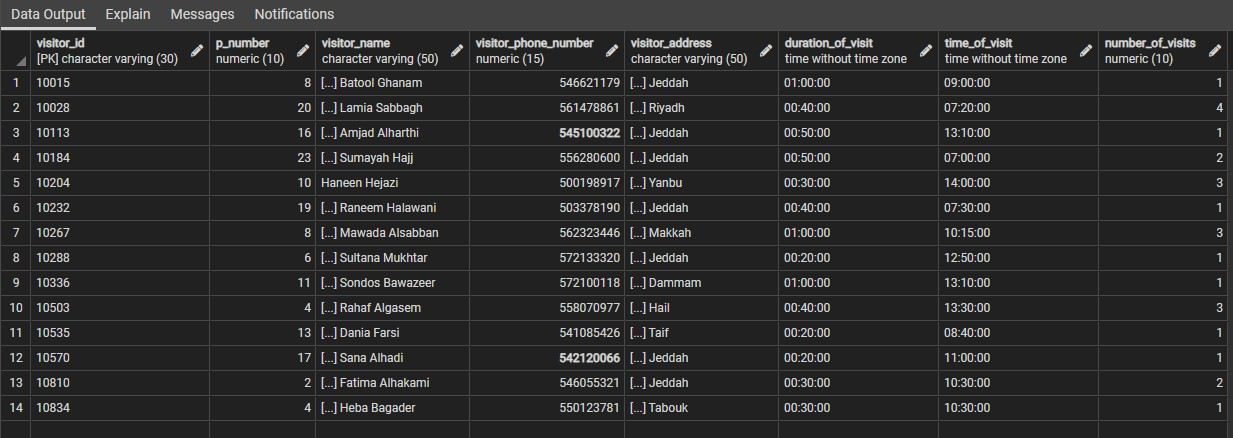
**Update in Natural Language (English)**

UPDATE the phone number of the visitor with ID (10015).

**SQL Script**



**Before Update:**



**After Update:**

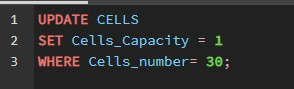


### 8.8.2 example

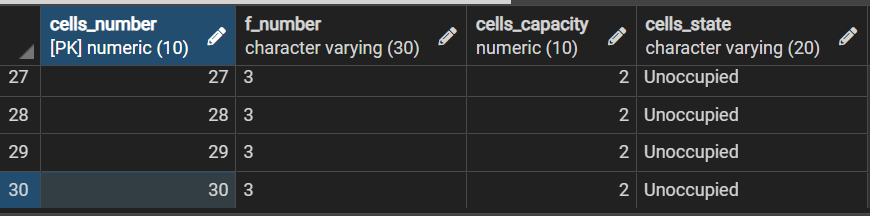
**Update in Natural Language (English)**

Update the cell’s capacity of the cell with number 30 from 2 to 1.

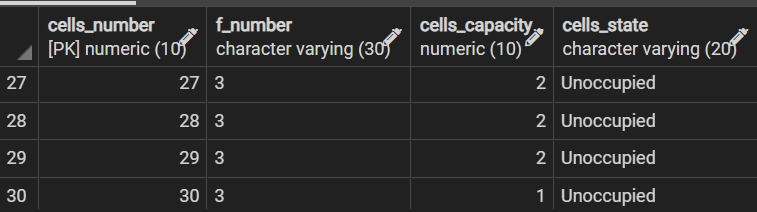
**SQL Script**



**Caption of the Output before update:**



**Caption of the Output after update:**



## 8.9 Delete Example

### 8.9.1 example

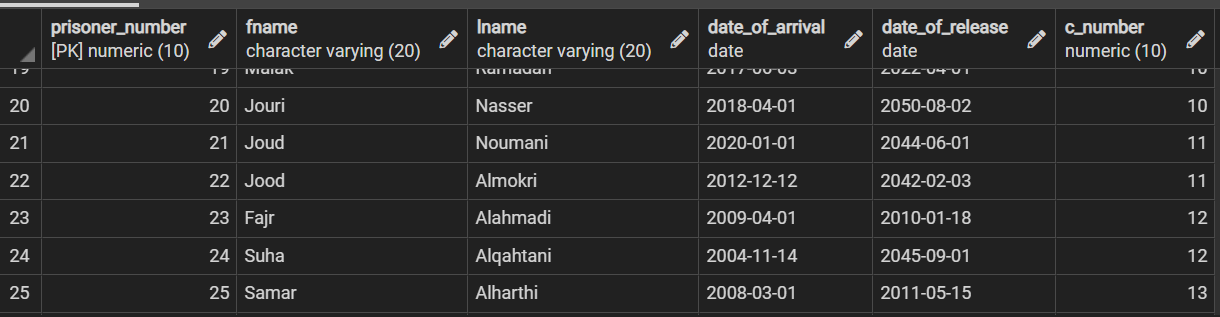
**Delete in Natural Language (English)**

Delete the prisoners with the number (00025).

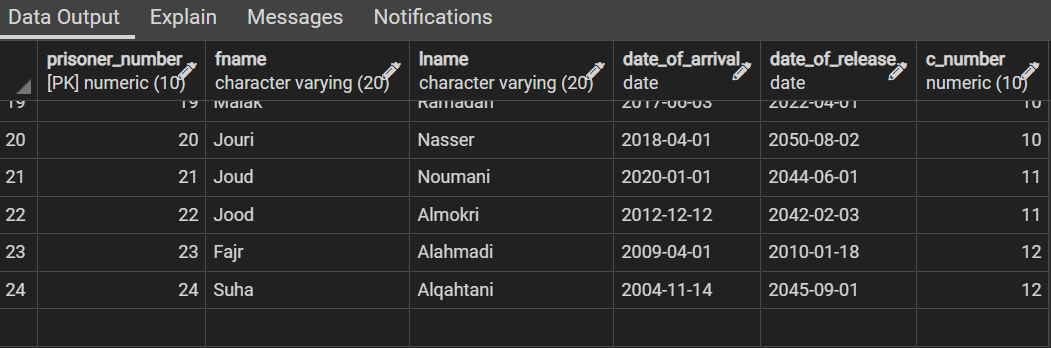
**SQL Script**



**Caption of the Output before delete:**



**Caption of the Output after delete:**



### 8.9.2 example

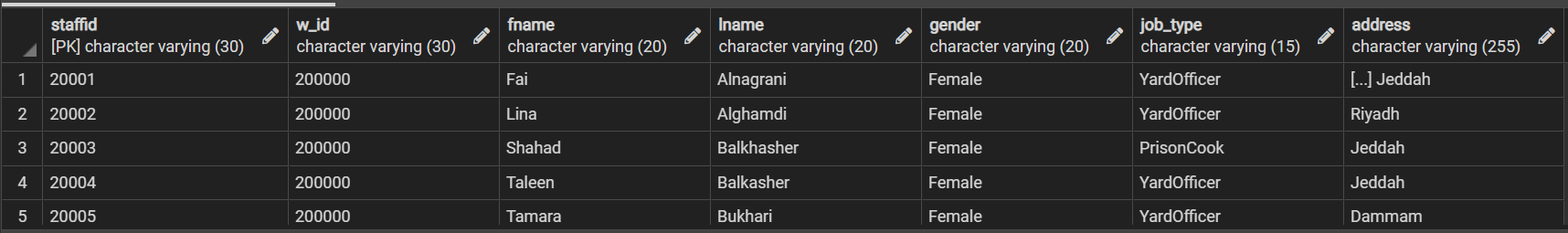
**Delete in Natural Language (English)**

Delete the STAFF member with the ID (20021).

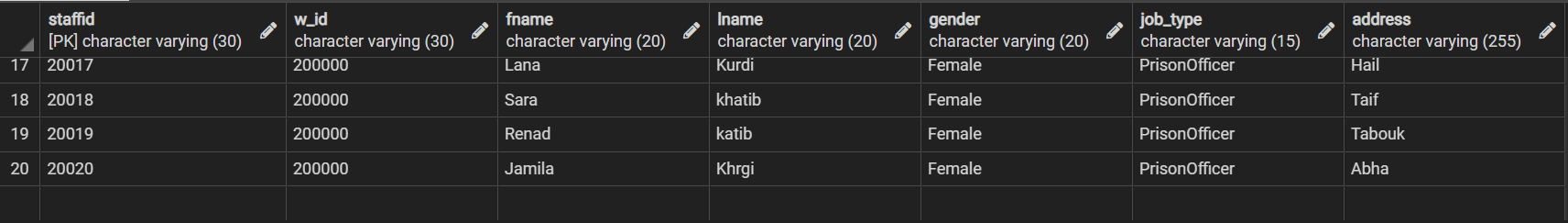
**SQL Script**



**Caption of the Output before delete:**



**Caption of the Output after delete:**

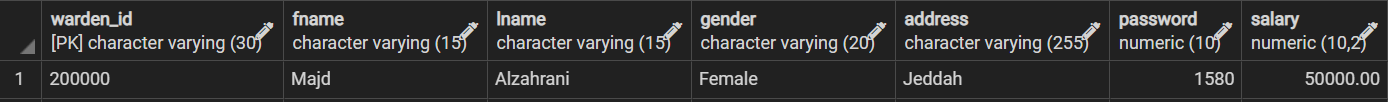


# 9 APPENDIX

Diagram, schematic

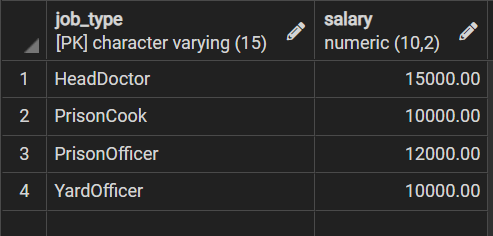
Description automatically generated

## WARDEN TABLE



## JOB\_SALARYTABLE

A picture containing text, monitor, computer, indoor

Description automatically generated

## STAFFTABLE

A picture containing text, monitor, road, black

Description automatically generated

**FLOOR TABLE**

****

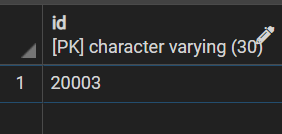
**PRISON\_OFFICERSTABLE**

Graphical user interface

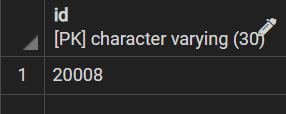
Description automatically generated with low confidenceTable

Description automatically generated with medium confidence

## PRISON\_COOKTABLE



**HEAD\_DOCTORTABLE**



## YARD\_OFFICERTABLE

A picture containing graphical user interface

Description automatically generatedTable

Description automatically generated

## CELLSTABLE

Table

Description automatically generated with medium confidence

A picture containing table

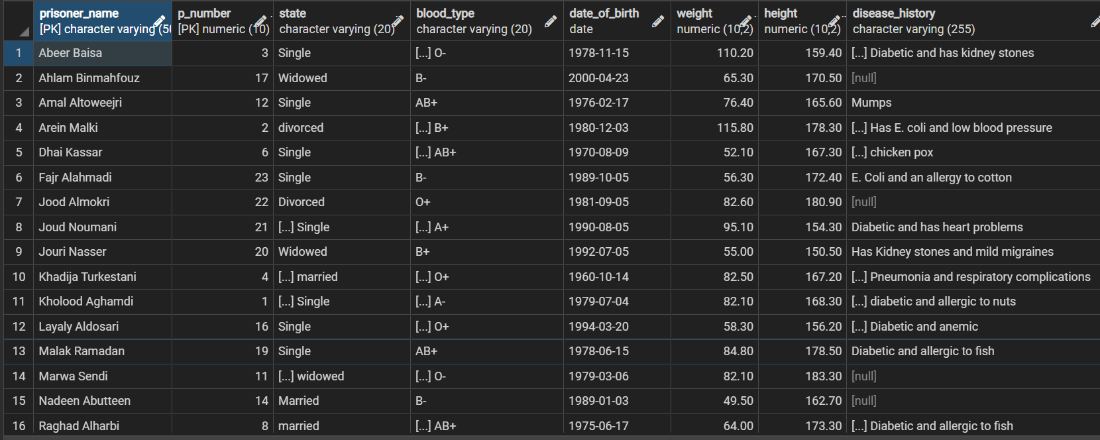
Description automatically generatedA picture containing table

Description automatically generated

## PRISONERSTABLE



## HEALTH\_RECORDSTABLE



## CRIMINAL\_RECORDSTABLE



## VISITSTABLE

A screenshot of a computer

Description automatically generated with medium confidence

## ACTIVITY\_REWARDTABLE

A screenshot of a computer

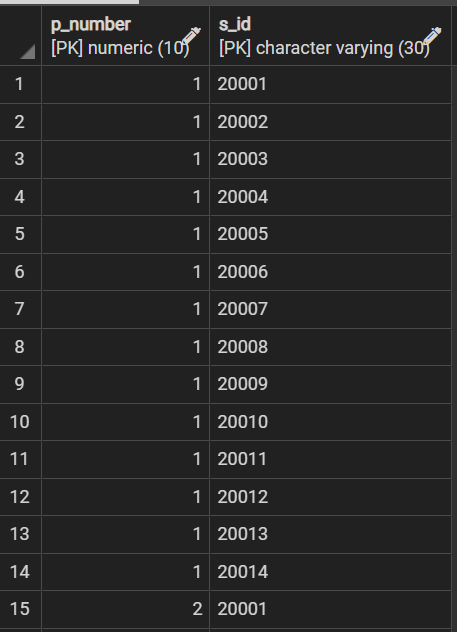
Description automatically generated with medium confidence

## ACTIVITIESTABLE

Table

Description automatically generated with medium confidence

## MANAGEDTABLE

Table

Description automatically generated with medium confidence

Table

Description automatically generated with medium confidence

And so on.

## CELLS\_FLOORTABLE

Table

Description automatically generated

## WARDEN\_FLOORTABLE

Graphical user interface, text

Description automatically generated

## PUNISHMENTTABLE

A screenshot of a computer

Description automatically generated with medium confidence

## LAW\_VIOLATORS TABLE

