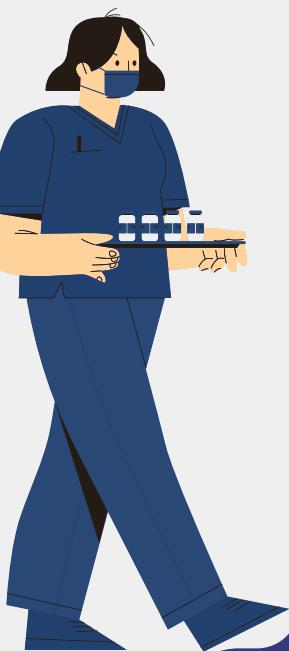


Stage 3 : Physical database implementation

DATA BASE



Vaccine Center Database

"Final Report"

Student's Name:

Abeer Almalawi 2006979

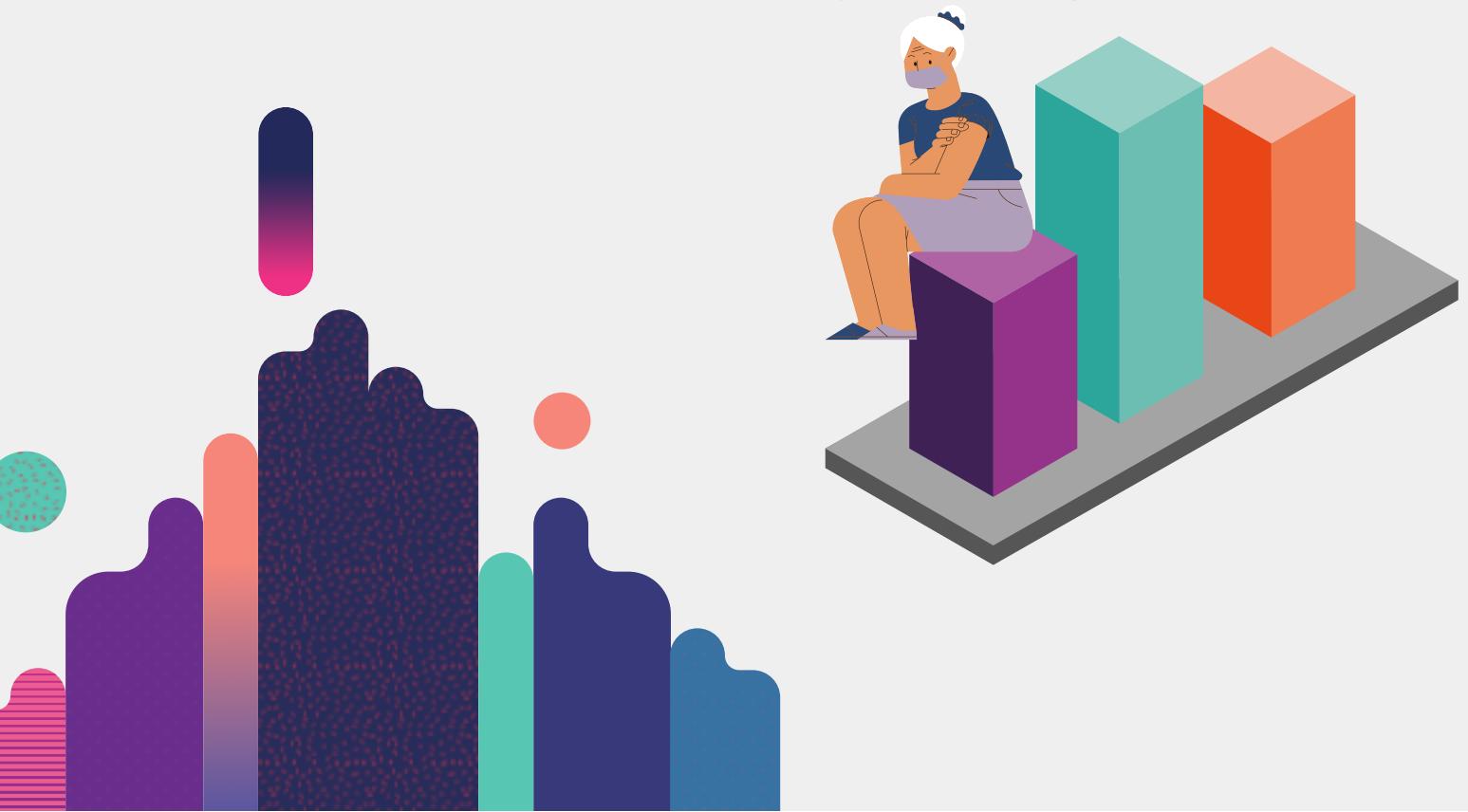
Raghad Bin Ali 2006601

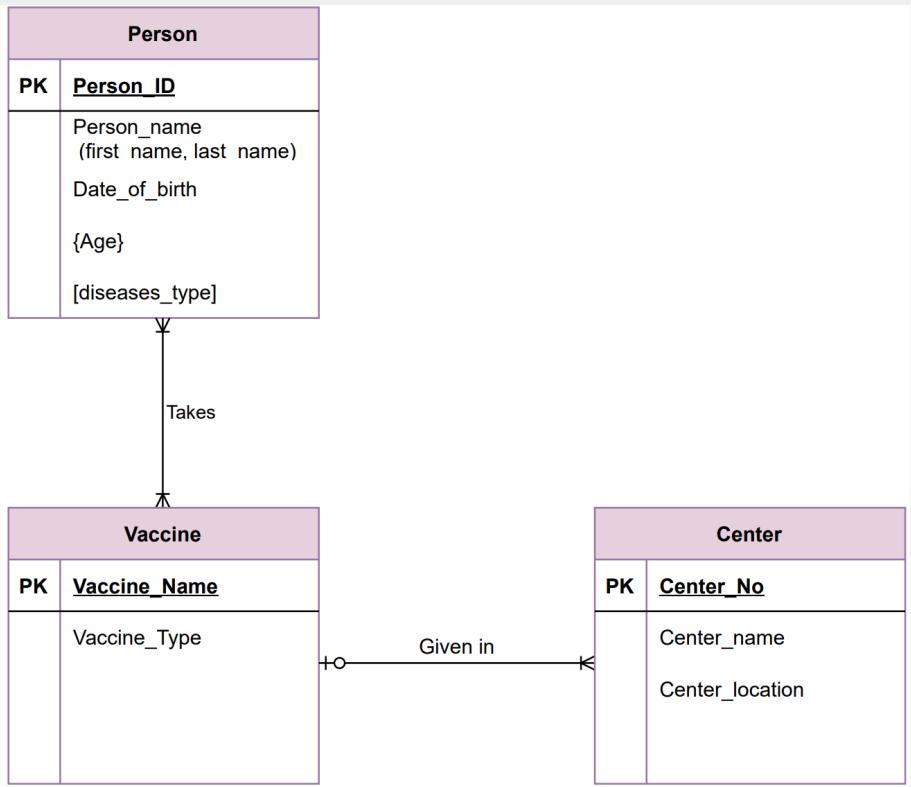
Maram Alharthi 2006492

Moudi Alhazza 2007279

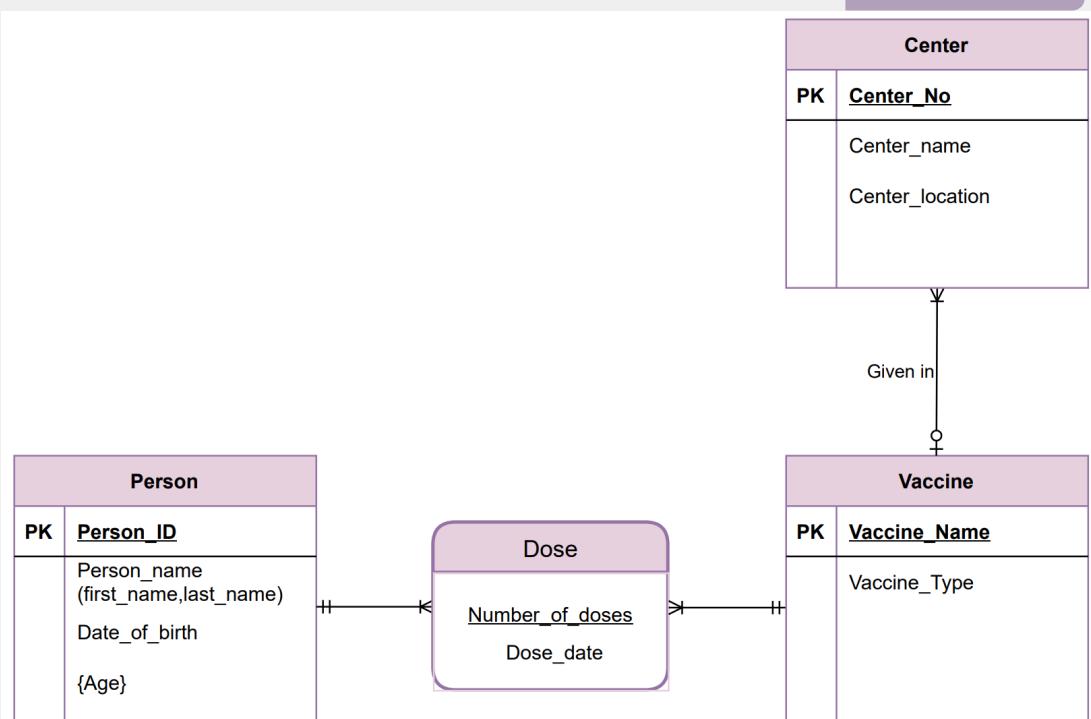
Our project is about a vaccine center's database, there are 3 entities which is : person, vaccine, and center . Each of them has it's own attributes and relationships .

- Each PERSON has an identifier ID-Number, Name "composed" of (FirstName, LastName), DateOfBirth, Age "derived" from DateOfBirth, and Disease_Type "multivalued" .
- Each VACCINE has an identifier Vaccine_Name and Vaccine_Type.
- Each CENTER has an identifier Center_Number, Center_Name and Location.
- Every PERSON must take one or more VACCINE , therefore each VACCINE is taken by one or more PERSON and this VACCINE is given in specific CENTER.





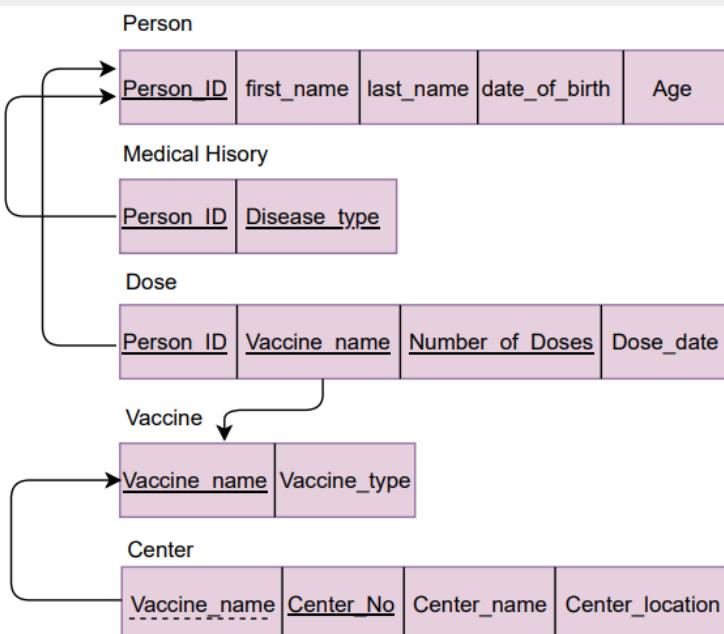
Entity Relationship Diagram



Entity Relationship Diagram After Handling M-N Relationship



RELATIONS SCHEMA :



NORMALIZATION :

PERSON (Person_ID, First_name, Last_name, Date_of_birth, Age)

MEDICAL HISTORY (Person_ID, Disease_Type)

DOSE (Person_ID, Vaccine_Name, Number_of_doses, Dose_Date)

VACCINE (Vaccine_Name, Vaccine_Type)

CENTER (Center_No, Center_Name, Center_location, Vaccine_Name)

*1NF :

- There is no repeating group .
- A unique key has been identified for each relation .

*2NF :

- All relations are in 1NF .
- All non-key attributes are fully functional dependency on the entire key.

*3NF :

- All relations are in 2NF .
- There is no transitive dependencies .

The above relations are all in 1NF, 2NF, and 3NF already .



Live SQL

SQL Worksheet

```
1 -----Tables Part (create & insert)
2 --First table "Person"
3 CREATE TABLE person (
4     person_ID NUMBER (7),
5     first_name VARCHAR2 (10),
6     last_name VARCHAR2 (10),
7     date_of_birth DATE,
8     age NUMBER (2),
9     CONSTRAINT person_pk PRIMARY KEY (person_ID));
10
11 insert into person(person_ID,first_name,last_name,date_of_birth,age) values(1,'Raghad','Ali','02-Dec-2002',19);
12 insert into person(person_ID,first_name,last_name,date_of_birth,age) values(2,'Abeer','Almalawi','26-Dec-2001',20);
13 insert into person(person_ID,first_name,last_name,date_of_birth,age) values(3,'Maram','Alharthi','18-Aug-1999',22);
14 insert into person(person_ID,first_name,last_name,date_of_birth,age) values(4,'Moudi','Alhazzaz','01-Apr-1996',25);
15 insert into person(person_ID,first_name,last_name,date_of_birth,age) values(5,'Aneen','Alghamdi','08-Oct-2003',18);
16
17 --Second table "Medical history"
18 CREATE TABLE medical_history (
19     disease_type VARCHAR2 (15),
20     fk_person NUMBER (7) REFERENCES person (person_ID),
21     CONSTRAINT medical_pk PRIMARY KEY (disease_type));
22
23 insert into medical_history (disease_type,fk_person) values('Pressure',1);
24 insert into medical_history (disease_type,fk_person) values('Diabetes',3);
25 insert into medical_history (disease_type,fk_person) values('Asthma',4).
```

Table created.

© 2021 Oracle Corporation · Privacy · Terms of Use
Oracle Learning Library · Ask Tom · Dev Gym · Database Documentation · Follow on Twitter
Live SQL 21.4.1, running Oracle Database 19c Enterprise Edition - 19.8.0.0
Built with ❤ using Oracle APEX running on Oracle Cloud Infrastructure and Oracle Kubernetes Engine

-----Part 2 (Create&insert)-----



Live SQL

SQL Worksheet

```
25 insert into medical_history (disease_type,fk_person) values('Asthma',4);
26 insert into medical_history (disease_type,fk_person) values('Influenza',5);
27 insert into medical_history (disease_type,fk_person) values('allergy',2);
28
29 --Third table "Vaccine"
30 CREATE TABLE vaccine (
31     vaccine_name VARCHAR2 (15),
32     vaccine_type VARCHAR2 (10),
33     CONSTRAINT vaccine_pk PRIMARY KEY (vaccine_name));
34
35 insert into vaccine(vaccine_name,vaccine_type) values('Pfizer','American');
36 insert into vaccine(vaccine_name,vaccine_type) values('Moderna','American');
37 insert into vaccine(vaccine_name,vaccine_type) values('SinoVac','Chinese');
38 insert into vaccine(vaccine_name,vaccine_type) values('SinoPharm','Chinese');
39 insert into vaccine(vaccine_name,vaccine_type) values('AstraZeneca','British');
40
41 --Fourth table "Center"
42 CREATE TABLE center (
43     center_no NUMBER (2),
44     center_name VARCHAR (30),
45     center_location VARCHAR2 (10),
46     CONSTRAINT center_pk PRIMARY KEY (center_no),
47     fk_vaccine VARCHAR2 (15) REFERENCES vaccine (vaccine_name));
48
49 insert into center(center_no,center_name,center_location,fk_vaccine) values(01,'KA Airport','Jeddah','Pfizer');
```

Table created.

© 2021 Oracle Corporation · Privacy · Terms of Use
Oracle Learning Library · Ask Tom · Dev Gym · Database Documentation · Follow on Twitter
Live SQL 21.4.1, running Oracle Database 19c Enterprise Edition - 19.8.0.0
Built with ❤ using Oracle APEX running on Oracle Cloud Infrastructure and Oracle Kubernetes Engine

-----Part 3 (Create&insert)-----



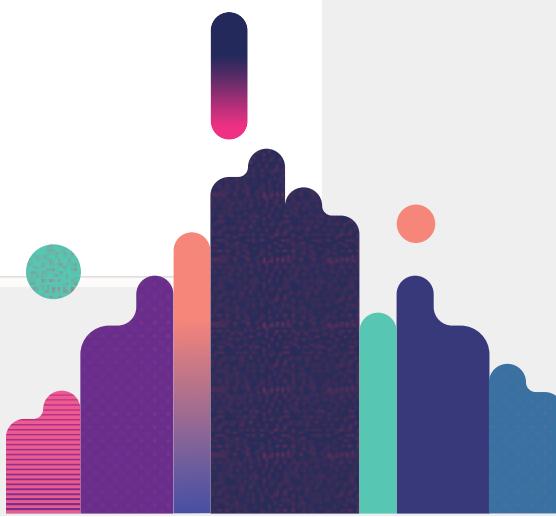
Live SQL

SQL Worksheet

```
45 center_location VARCHAR2 (10),
46 CONSTRAINT center_pk PRIMARY KEY (center_no),
47 fk_vaccine VARCHAR2 (15) REFERENCES vaccine (vaccine_name));
48
49 insert into center(center_no,center_name,center_location,fk_vaccine) values(01,'KA Airport','Jeddah','Pfizer');
50 insert into center(center_no,center_name,center_location,fk_vaccine) values(02,'KingAbdullah Hospital','Riyadh','Modrena');
51 insert into center(center_no,center_name,center_location,fk_vaccine) values(03,'KingFahad Hospital','Jeddah','Modrena');
52 insert into center(center_no,center_name,center_location,fk_vaccine) values(04,'KingFaisal Hospital','Dammam','Pfizer');
53 insert into center(center_no,center_name,center_location,fk_vaccine) values(05,'KingAbdullah Hospital','Makkah','AstraZeneca');
54
55 --Fifth table "Dose"
56 CREATE TABLE dose (
57 number_of_doses NUMBER (1),
58 dose_date DATE,
59 fk_vaccine VARCHAR2 (15) REFERENCES vaccine (vaccine_name),
60 fk_person NUMBER (7) REFERENCES person (person_ID),
61 CONSTRAINT dose_pk PRIMARY KEY (number_of_doses));
62
63 insert into dose(number_of_doses,dose_date,fk_vaccine,fk_person) values(4,'09-Oct-2020','SinoPharm',1);
64 insert into dose(number_of_doses,dose_date,fk_vaccine,fk_person) values(5,'15-Mar-2020','SinoVac',2);
65 insert into dose(number_of_doses,dose_date,fk_vaccine,fk_person) values(1,'23-Apr-2021','Modrena',3);
66 insert into dose(number_of_doses,dose_date,fk_vaccine,fk_person) values(3,'18-Nov-2019','Pfizer',4);
67 insert into dose(number_of_doses,dose_date,fk_vaccine,fk_person) values(2,'10-Dec-2019','AstraZeneca',5);
68
69 commit;
```

Table created.

Outputs



Live SQL

SQL Worksheet

```
Table created.

1 row(s) inserted.

Table created.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.
```

-----Outputs (part 2)-----

The screenshot shows the 'Live SQL' interface with a sidebar and a main workspace. The sidebar includes links for Home, SQL Worksheet (which is selected), My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main workspace is titled 'SQL Worksheet' and displays the following output:

```
1 row(s) inserted.  
1 row(s) inserted.  
Table created.  
1 row(s) inserted.  
1 row(s) inserted.  
1 row(s) inserted.  
1 row(s) inserted.  
Table created.  
1 row(s) inserted.
```

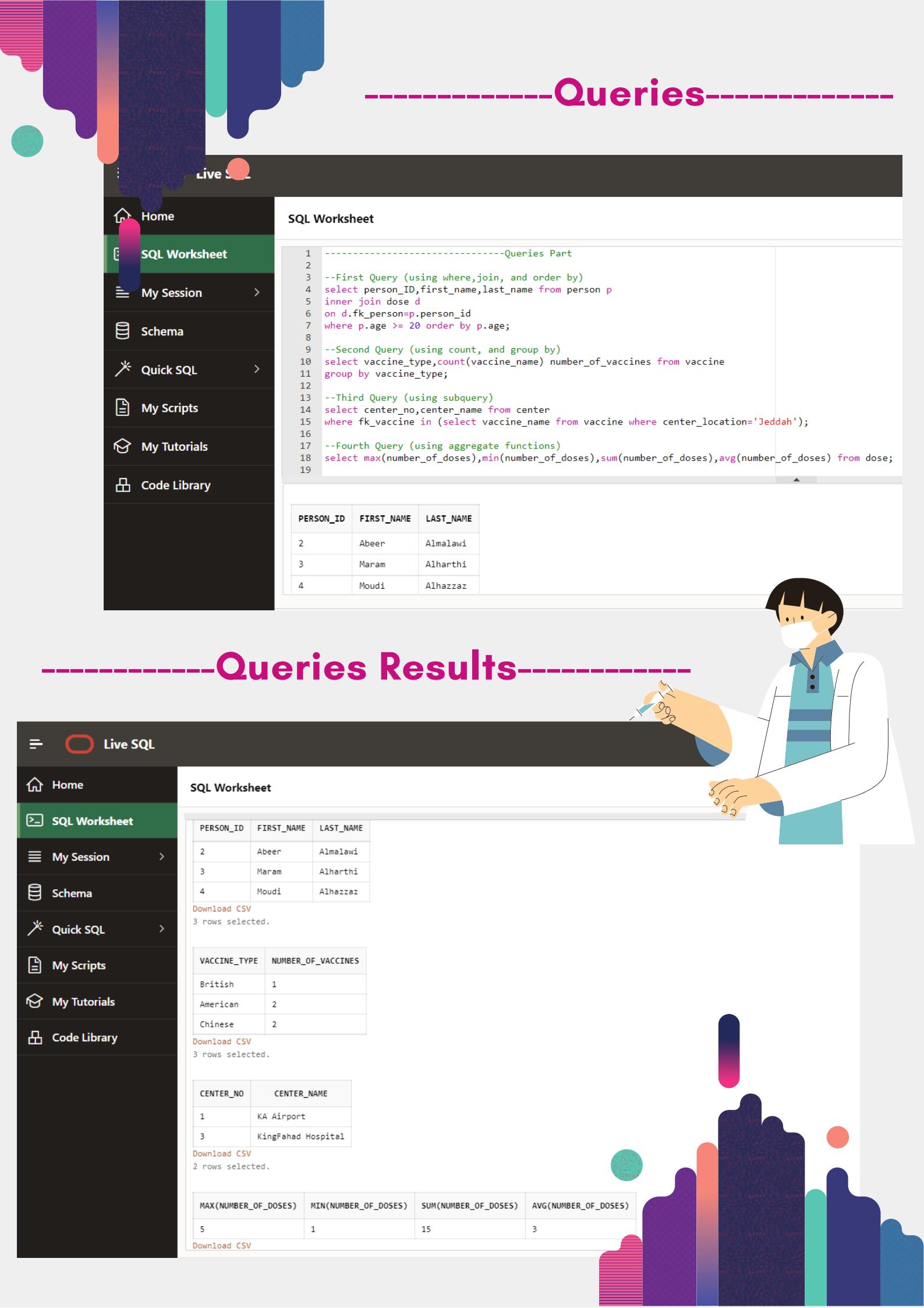


-----Outputs (part 3)-----

The screenshot shows the 'Live SQL' interface with a sidebar and a main workspace. The sidebar includes links for Home, SQL Worksheet (which is selected), My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main workspace is titled 'SQL Worksheet' and displays the following output:

```
1 row(s) inserted.  
1 row(s) inserted.  
1 row(s) inserted.  
1 row(s) inserted.  
Table created.  
1 row(s) inserted.  
1 row(s) inserted.  
1 row(s) inserted.  
1 row(s) inserted.  
1 row(s) inserted.
```

Queries



Live SQL

Home

SQL Worksheet

My Session >

Schema

Quick SQL >

My Scripts

My Tutorials

Code Library

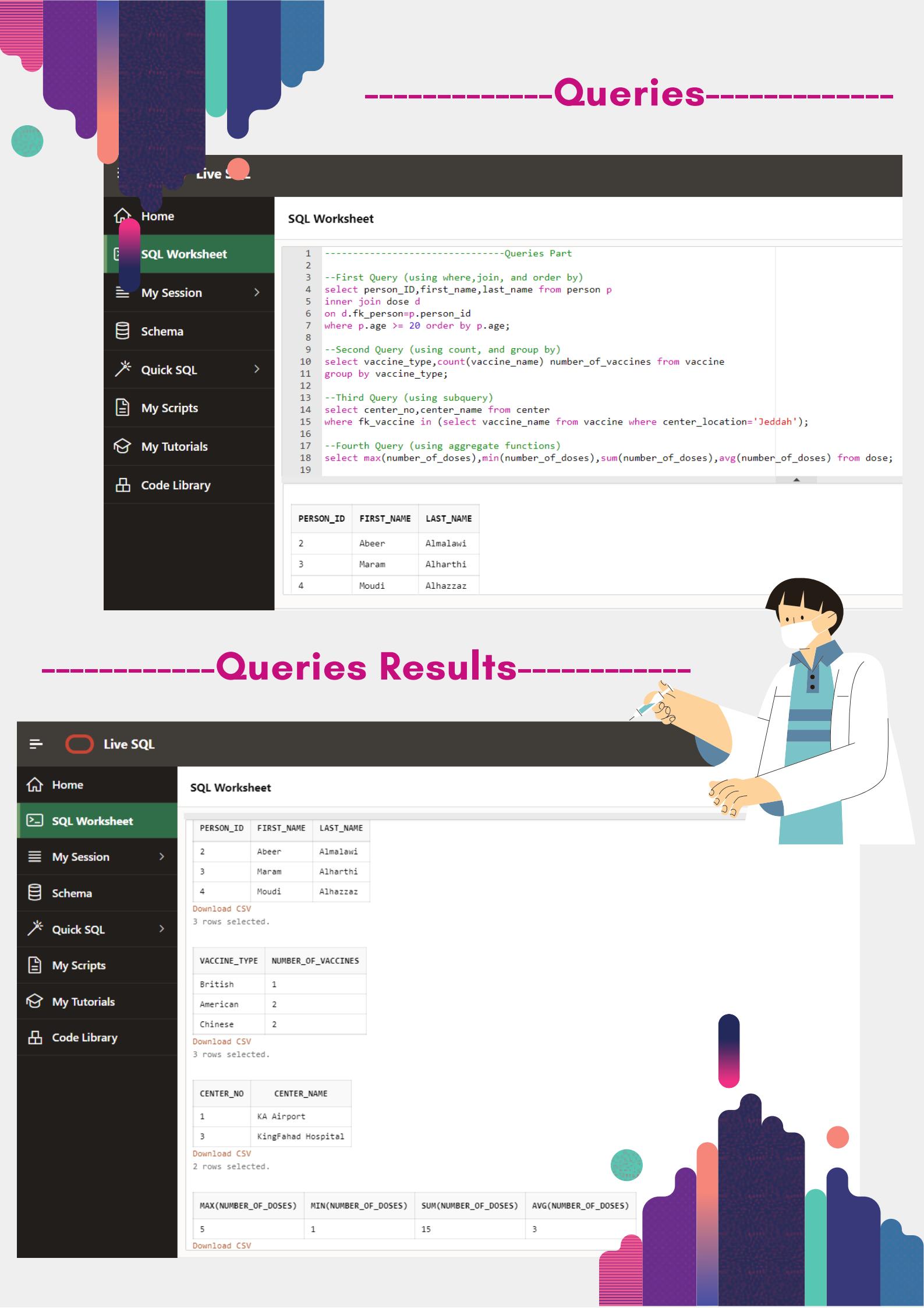
SQL Worksheet

```
1 -----Queries Part
2
3 --First Query (using where,join, and order by)
4 select person_ID,first_name,last_name from person p
5 inner join dose d
6 on d.fk_person=p.person_id
7 where p.age >= 20 order by p.age;
8
9 --Second Query (using count, and group by)
10 select vaccine_type,count(vaccine_name) number_of_vaccines from vaccine
11 group by vaccine_type;
12
13 --Third Query (using subquery)
14 select center_no,center_name from center
15 where fk_vaccine in (select vaccine_name from vaccine where center_location='Jeddah');
16
17 --Fourth Query (using aggregate functions)
18 select max(number_of_doses),min(number_of_doses),sum(number_of_doses),avg(number_of_doses) from dose;
19
```

PERSON_ID	FIRST_NAME	LAST_NAME
2	Abeer	Almalawi
3	Maram	Alharthi
4	Moudi	Alhazzaz



Queries Results



Live SQL

Home

SQL Worksheet

My Session >

Schema

Quick SQL >

My Scripts

My Tutorials

Code Library

SQL Worksheet

PERSON_ID	FIRST_NAME	LAST_NAME
2	Abeer	Almalawi
3	Maram	Alharthi
4	Moudi	Alhazzaz

Download CSV

3 rows selected.

VACCINE_TYPE	NUMBER_OF_VACCINES
British	1
American	2
Chinese	2

Download CSV

3 rows selected.

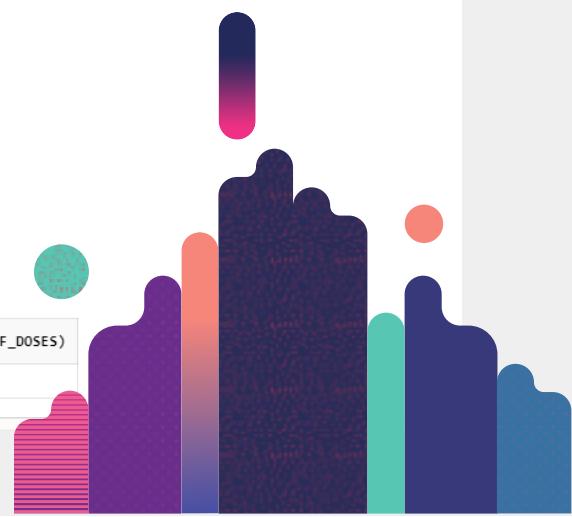
CENTER_NO	CENTER_NAME
1	KA Airport
3	KingFahad Hospital

Download CSV

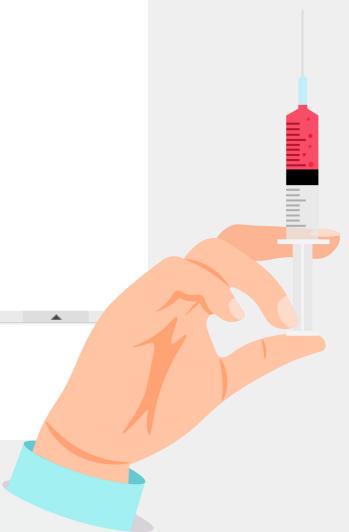
2 rows selected.

MAX(NUMBER_OF_DOSSES)	MIN(NUMBER_OF_DOSSES)	SUM(NUMBER_OF_DOSSES)	AVG(NUMBER_OF_DOSSES)
5	1	15	3

Download CSV



-----Procedures 1-----

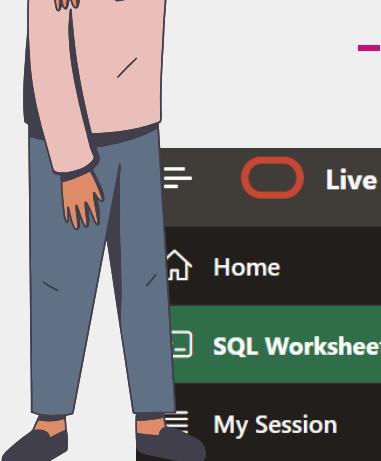


A screenshot of a SQL worksheet interface. On the left is a sidebar with options: Home, SQL Worksheet (which is selected), My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main area shows a code editor with the following PL/SQL procedure:

```
1 -----Procedures part
2 --First procedure
3 create or replace procedure person_doses (per_id int)
4 is
5 p_id number;
6 f_name varchar(10);
7 l_name varchar(10);
8 age number;
9 no_of_doses number;
10 begin
11 p_id :=null;
12 f_name :=null;
13 l_name :=null;
14 age :=null;
15 no_of_doses :=null;
16
17 select person_ID,first_name,last_name,age,number_of_doses
18 into p_ID,f_name,l_name,age,no_of_doses
19 from person p
20 inner join dose d
21 on d.fk_person=p.person_id
22 where p.person_id=per_id;
23
24 dbms_output.put_line(p_ID || ',' || f_name || ',' || l_name || ',' || age || ',' || no_of_doses);
25 end;
```

Below the code, the message "Procedure created." is displayed.

-----Procedure 1 Results-----



A screenshot of a SQL worksheet interface. On the left is a sidebar with options: Home, SQL Worksheet (selected), My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main area shows a code editor with the following PL/SQL statement:

```
1 execute person_doses(1);
```

Below the code, the message "Statement processed." is followed by the output: "1,Raghad,Ali,19,4".

-----Procedure 2-----



Live SQL

- Home
- SQL Worksheet**
- My Session >
- Schema
- Quick SQL >
- My Scripts
- My Tutorials
- Code Library

SQL Worksheet

```
1 --Second procedure
2 create or replace procedure person_name(f_name varchar2,p_id int)
3 is
4 begin
5     UPDATE person SET first_name = f_name where person_id = p_id;
6     commit;
7 end;|
```

Procedure created.

-----Procedure 2 Results-----



Live SQL

- Home
- SQL Worksheet**
- My Session >
- Schema
- Quick SQL >
- My Scripts
- My Tutorials
- Code Library

SQL Worksheet

```
1 execute person_name('Sara',3);
2
3 select * from person |
```

Statement processed.

PERSON_ID	FIRST_NAME	LAST_NAME	DATE_OF_BIRTH	AGE
1	Raghad	Ali	02-DEC-02	19
2	Abeer	Almalawi	26-DEC-01	20
3	Sara	Alharthi	18-AUG-99	22
4	Moudi	Alhazzaz	01-APR-96	25
5	Aneer	Alghamdi	08-OCT-03	18

[Download CSV](#)
5 rows selected.