



**Green University of Bangladesh**  
**Department of Computer Science and Engineering (CSE)**  
**Faculty of Sciences and Engineering**  
**Semester: (Spring, Year:2021), B.Sc. in CSE (Day/Eve)**

**Course Title: Database System Lab**  
**Course Code: CSE 210                      Section: DJ**

**Lab Project Name: Vaccination data analysis on a university database with SQL.**

**Student Details**

<b>Name</b>	<b>ID</b>
Abdullah Al Fahad	201002037

**Submission Date                      : 15/5/2022**  
**Course Teacher's Name              : Ms. Mahmuda Rahman**

**Lab Project Status**

**Marks: .....**

**Signature: .....**

**Comments: .....**

**Date: .....**

# Table of Contents

<b>Chapter 1 Introduction .....</b>	<b>3</b>
1.1 Introduction .....	3
1.2 Design Goals/Objective .....	3
<b>Chapter 2 .....</b>	<b>4</b>
<b>Design .....</b>	<b>4</b>
2.1 Tables .....	4
2.2 Attributes.....	4
.....	4
<b>Chapter 3 .....</b>	<b>7</b>
<b>Performance Evaluation .....</b>	<b>7</b>
3.1 Triggers .....	7
3.2 Results and Discussions .....	15
<b>Chapter 4 Conclusion .....</b>	<b>17</b>
4.1 Introduction .....	17
4.1 Practical Implications .....	17
4.2 Scope of Future Work .....	17

# Chapter 1

## Introduction

### 1.1 Introduction

This project is a database for storing all kinds of vaccination related data of large-scale institution like universities, educational organizations. This database will help big enterprises to keep track of the current vaccination status of their respective institutions. To make big decisions and to minimize the spread of covid the data collected with this database can come in handy in tough situations.

It is not just a database for storing data regarding vaccination. Many research-based data analyzation can be made through sql queries as well through this database.

### 1.2 Design Goals/Objective

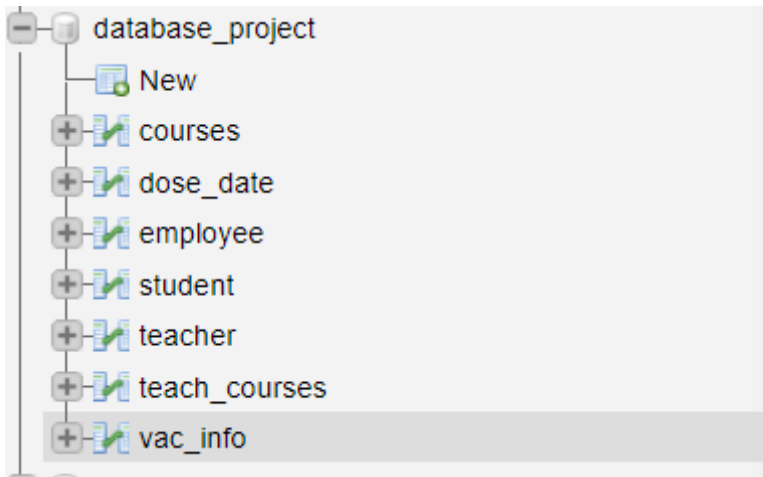
The objectives for this project are stated below:

- Data Analyzation
- Gathering various statistics regarding the issue.
- To help manage vaccination related data in a sql database.
- To gather various vaccination related information by checking various parameters.

# Chapter 2

## Design

### 2.1 Tables



### 2.2 Attributes

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	grades	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change
<input type="checkbox"/> 2	course_code	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change
<input type="checkbox"/> 3	status	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change
<input type="checkbox"/> 4	credit	int(11)			Yes	NULL			Change
<input type="checkbox"/> 5	section	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change
<input type="checkbox"/> 6	semester	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change
<input type="checkbox"/> 7	stu_id	int(11)			Yes	NULL			Change

Courses table

#	Name	Type	Collation	Attributes	Null	Default	Comments
<input type="checkbox"/> 1	emp_id	int(11)			Yes	NULL	
<input type="checkbox"/> 2	stu_id	int(11)			Yes	NULL	
<input type="checkbox"/> 3	date	datetime			No	None	
<input type="checkbox"/> 4	name_vac	varchar(255)	utf8mb4_general_ci		No	None	

Dose\_date table

#	Name	Type	Collation	Attributes	Null	Default	Comments
<input type="checkbox"/> 1	Gender	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 2	Position	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 3	Email	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 4	Campus_location	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 5	Salary	int(11)			Yes	NULL	
<input type="checkbox"/> 6	Phone_no	int(11)			Yes	NULL	
<input type="checkbox"/> 7	Address	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 8	Blood_group	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 9	emp_id	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 10	name	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 11	age	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 12	DOB	datetime			No	None	

Employee table

#	Name	Type	Collation	Attributes	Null	Default	Comment
<input type="checkbox"/> 1	name	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 2	email	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 3	stu_id	int(11)			Yes	NULL	
<input type="checkbox"/> 4	current_sem	int(11)			Yes	NULL	
<input type="checkbox"/> 5	address	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 6	cgpa	double			Yes	NULL	
<input type="checkbox"/> 7	gender	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 8	deparment	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 9	Phone_no	int(11)			Yes	NULL	
<input type="checkbox"/> 10	campus	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 11	blood_group	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 12	credits_completed	int(11)			Yes	NULL	
<input type="checkbox"/> 13	advisor	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 14	AGE	varchar(255)	utf8mb4_general_ci		Yes	NULL	
<input type="checkbox"/> 15	DOB	datetime			No	None	

Student table

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Ext
<input type="checkbox"/>	1	name	varchar(255)	utf8mb4_general_ci		Yes	NULL		
<input type="checkbox"/>	2	course_code	varchar(255)	utf8mb4_general_ci		Yes	NULL		

Teacher table

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	.
<input type="checkbox"/>	1	stu_id	int(11)			Yes	NULL			
<input type="checkbox"/>	2	emp_id	int(11)			Yes	NULL			
<input type="checkbox"/>	3	status	varchar(255)	utf8mb4_general_ci		Yes	NULL			
<input type="checkbox"/>	4	booster	varchar(255)	utf8mb4_general_ci		Yes	NULL			
<input type="checkbox"/>	5	no_of_doses	int(11)			No	None			
<input type="checkbox"/>	6	name_vac	varchar(255)	utf8mb4_general_ci		No	None			

Vac\_info table

# Chapter 3

## Performance Evaluation

### 3.1 Triggers

**T1.**

Table: student

Time: After

Event: Delete

**Def:**

BEGIN

DELETE FROM

vac\_info WHERE

stu\_id=old.stu\_id;

END

**T2.**

Table: student

Time: before

Event: insert

**Def:**

BEGIN

insert into

vac\_info(stu\_id)

VALUES(new.stu\_id);

UPDATE vac\_info

SET

status = "incomplete" where stu\_id = new.stu\_id;

if(new.dob) THEN

SET new.age=(DATE\_FORMAT(FROM\_DAYS(DATEDIFF(now(),new.DOB)), '% Y')+0);

end if;

end

**T3.**

Table: dose\_date

Time: after

Event: delete

**Def:**

BEGIN

UPDATE vac\_info SET no\_of\_doses =

(SELECT COUNT(\*))

FROM dose\_date

where



```
emp_id= old.emp_id)
```

```
WHERE emp_id=old.emp_id;
```

```
UPDATE vac_info SET no_of_doses =
```

```
(SELECT COUNT(*)
```

```
FROM dose_date
```

```
where
```

```
stu_id= old.stu_id)
```

```
WHERE stu_id = old.stu_id;
```

```
END
```

#### **T4.**

Table: Employee

Time: before

Event: insert

#### **Def:**

```
BEGIN
```

```
if(new.position='teacher')
```

```
THEN
```

```
insert into
```

```
teacher(emp_id)
```

```
VALUES (new.emp_id);
```

```
end if;
```

```
insert into
```

```

vac_info(emp_id)

VALUES (new.emp_id);

UPDATE vac_info

SET

status = "incomplete" where emp_id = new.emp_id;

if(new.dob) THEN

SET new.age=(DATE_FORMAT(FROM_DAYS(DATEDIFF(now(),new.DOB)), '%Y')+0);

end if;

end

```

## **T5.**

Table: Employee

Time: after

Event: delete

### **Def:**

```

BEGIN

DELETE FROM teacher WHERE emp_id= OLD.emp_id;

DELETE FROM vac_info WHERE emp_id= OLD.emp_id;

END

```

## **T6.**

Table: dose\_date

Time: after

Event: insert

**Def:**

BEGIN

DECLARE a int;

DECLARE b int;

SELECT count(\*) into a from dose\_date where emp\_id= new.emp\_id;

SELECT count(\*) into b from dose\_date where stu\_id= new.stu\_id;

UPDATE vac\_info SET no\_of\_doses =

(SELECT COUNT(\*))

FROM dose\_date

where

emp\_id= new.emp\_id)

WHERE emp\_id=new.emp\_id;

UPDATE vac\_info SET no\_of\_doses =

(SELECT COUNT(\*))

FROM dose\_date

where

stu\_id= new.stu\_id)

WHERE stu\_id = new.stu\_id;

```
if(a>=2) THEN
```

```
UPDATE vac_info
```

```
SET status="complete" where emp_id = new.emp_id;
```

```
end if;
```

```
if(a=1) THEN
```

```
UPDATE vac_info
```

```
SET status="partial" where emp_id = new.emp_id;
```

```
end if;
```

```
if(a<1) THEN
```

```
UPDATE vac_info
```

```
SET status="incomplete" where emp_id = new.emp_id;
```

```
end if;
```

```
if(a>=2) THEN
```

```
UPDATE vac_info
```

```
SET status="complete" where stu_id = new.stu_id;
```

```
end if;
```

```
if(a=1) THEN
```

```
UPDATE vac_info
```

```
SET status="partial" where stu_id = new.stu_id;
```

end if;

if(a=0) THEN

UPDATE vac\_info

SET status="incomplete" where stu\_id = new.stu\_id;

end if;

UPDATE vac\_info SET name\_vac = new.name\_vac

WHERE stu\_id = new.stu\_id;

UPDATE vac\_info SET name\_vac = new.name\_vac

WHERE emp\_id = new.emp\_id;

if(a>2) THEN

UPDATE vac\_info

SET booster="done" where stu\_id = new.stu\_id;

end if;

if(a>2) THEN

UPDATE vac\_info

SET booster="done" where emp\_id = new.emp\_id;

end if;

END

Queries:

**To find the number of brands of vaccine used in a uni:**

```
SELECT COUNT(*), name_vac FROM dose_date GROUP BY name_vac ORDER BY COUNT(*) DESC
```

**To count the number of student vaccinated in a class:**

Select

```
Select courses.course_code, courses.stu_id, vac_info.status from courses inner join vac_info on
```

```
courses.stu_id = vac_info.stu_id WHERE course_code='cse 310' and courses.status != 'complete';
```

```
Select count(courses.stu_id), courses.course_code, courses.stu_id, vac_info.status from courses inner join vac_info
```

```
on courses.stu_id = vac_info.stu_id WHERE course_code='cse 310' and courses.status != 'complete'
```

```
group by courses.course_code;
```

**Query to find if a teacher of class is vaccinated or not:**

```
SELECT teach_courses.name, teach_courses.course_code, vac_info.emp_id, vac_info.status FROM  
((employee INNER JOIN vac_info ON employee.emp_id = vac_info.emp_id) INNER JOIN teach_courses
```

```
ON teach_courses.name = employee.name);
```

### Query to find the number of student vaccinated in permanent campus

```
SELECT count(vac_info.stu_id), student.campus FROM vac_info inner join student on  
student.stu_id=vac_info.stu_id WHERE student.campus='permanent' and vac_info.status="complete";
```

### Query to find the students vaccinated in permanent campus

```
SELECT countvac_info.stu_id, student.campus FROM vac_info inner join student on  
student.stu_id=vac_info.stu_id WHERE student.campus='permanent' and vac_info.status="complete";
```

## 3.2 Results and Discussions

### 3.2.1 Results

Q1.

COUNT(*)	name_vac
2	cinopharm
1	pfizer

Q2.

+ Options		
course_code	stu_id	status
cse 310	201002037	complete

### Q3.

+ Options

name	course_code	emp_id	status
teacher1	cse 105	1	partial
teacher2	cse103	2	incomplete

Other queries and triggers are working as their supposed to as well.



# Chapter 4

## Conclusion

### 4.1 Introduction

This project's main aim is to help data analyst to analyze data and put forward statistics and get some kind valuable discovery out of the any type of research purpose.

### 4.1 Practical Implications

This project has many practical implications as well as it can be used as a database for storing vaccinated related data on institutes like universities.

### 4.2 Scope of Future Work

It is unlimited potential as many different types of combination in sql can be used to get different types of data and the space can increased to store more information.

