



Final_DB_Project_Submission

for

Vaccination drive database

Prepared by: S2_T10

DAIICT

Id	Name
201901274	Soham Patel
201901043	Nakul Patel
201901224	Shrey Mavani
201901259	Divyesh Patel

Contents

SECTION1: FINAL VERSION OF SRS.....	3
SECTION2: NOUN ANALYSIS.....	25
SECTION3: ER-DIAGRAMS ALL VERSIONS	32
SECTION4: CONVERSION OF FINAL ER-DIAGRAM TO RELATIONAL MODEL	34
SECTION5: NORMALIZATION AND SCHEMA REFINEMENT	36
SECTION6: FINAL DDL SCRIPTS, INSERT STATEMENTS, 40 SQL QURIES WITH SNAPSHOTS OF OUTPUT OF EACH QUERY	46
SECTION7: PROJECT CODE WITH OUTPUT SCREENSHOTS	95

SECTION1: FINAL VERSION OF SRS

INTRODUCTION:

➤ Purpose:

- This document is to build a Web Publishing System to manage and ease the process which helps to regulate vaccine centers. The purpose for this SRS documentation is to help developer to understand the system and its features. To make clients and employees more familiar with the system and even for managers to ease his tasks.
- The system provides an interface between consumer and employee of the vaccination centers, employee and vaccination center manager, vaccination center manager and manufacturers. This documentation helps developers to design a database that “Who has control to check if a particular vaccine is available”, “Which vaccine should be ordered?”, “How many doses are completed for a particular consumer?”, etc.

This document is intended to ease the understanding of the system for all users from customers to managers to manufacturer.

➤ Documentation Format:

- The format of this document is according to the referenced IEEE template for System Requirement Specifications (SRS) documents. The font convention is ‘Cooper Black’ for headings and sub-headings, and ‘Arial’ for other points. Use of bold words and underlined words are made for emphasizing important words which holds special significance in document. The document is divided in main topics followed by their related sub-topics which contains texts/images and description in it.

➤ Intended audience and Reading suggestions:

- The intended audience who will be using this document and software are classified as Clients, Employees, Managers, Manufacturers and Developers.
- The document is divided into sections which are introduction, overall description, background reading, list requirement, operating environment, product function, privileges, assumptions, and business constraints.
- Customers and employees need to refer mainly to list requirements and product function section.
- Managers and developers need to go through whole document as it will help them understand the system fully.

➤ Product scope:

- This software will be a system that will be used to store data and information regarding vaccination of general vaccines. The focus of the software is to ease the management of vaccine drives for different vaccines, create awareness and easy-to-use software for users.
- The system is designed to maximize the productivity of vaccination drives. It is helpful to store all the large data of customers, vaccines, availability of vaccines etc. which otherwise would have to be performed manually and takes lots of effort. By use of this system, it will be easy for customers also as they can find required vaccines and book appointments from home only.
- The system will be based on the relational database between vaccine centers, suppliers of vaccines and consumers. This software will allow concerned people to manage and update the data, so it will be secured.
- So, the workload of people associated with this drive will be minimized and the productivity, efficiency and awareness of vaccination drive will maximize. It will make it easy to understand the whole process as everything will be well defined and as access to data will be provided to concerned people.

Above all, we hope that this system provides an easy-to-use experience to all users using the system.

➤ Workflow description:

- In current system for vaccination drives, the basic workflow is like:
 1. Customers will book vaccine slots online from website. They can even do payment online on the website.
 2. Then after, at the vaccine center the employees will verify customer details and then after vaccinating customer they update the database and so then after customer can download certificate for being vaccinated.
 3. Task of manager and manufacturer is to keep track on availability of vaccines and try to avoid shortage of vaccines.

➤ References:

1. <https://github.com/Corona-Vaccine-Management-System/Corona-Management>
2. <https://www.cowin.gov.in>
3. <https://myhistoryfeed.medium.com/complete-software-requirements-specification-srs-example-5bd08f107206>
4. <https://www.who.int/emergencies/diseases/novel-coronavirus-2019/covid-19-vaccines>

OVERALL DESCRIPTION:

➤ Product perspective:

- Earlier vaccination was done manually and so it was hard to keep all the records of vaccinated people all over the country, so this new software provides a great help as it is a fully functional automated software which will replace all the manual efforts needed and this software will also ensure that there will be a significant improvement in the reliability, efficiency, performance, and security in the vaccination drive.
- The use of this software and database will provide a great comfort to store the user details, employee details, supplier's details availability of vaccines at centers, searching of user's data, ease in maintaining counts of vaccines taken, etc.... will be easy to manage and so by use of it we can even enhance the level of security of database as we can provide limited access to functions according to the user classes.

➤ User Function:

- Customers must be able to search their nearby vaccination center location.
- Customers can book an appointment at vaccine centers and even do online payment.

- Customers can download their vaccination certificate and even can apply for correction of certificate from the website.
- Employees(staff) must be able to verify details of customers like payment and age.
- Manager must be able to update new stocks of vaccines.
- Manager must be able to keep track of employees working in his center for vaccine drives.

➤ User classes and characteristics:

- Users who are expected to use the website are recognized into 5 main categories which are clients, employees, managers, manufacturer, and developers.
 1. Clients: Users who want to take vaccinations.
 2. Employees: Users who work at vaccination centers for vaccine drives like doctors, nurses, and other staff.
 3. Managers: Head of each vaccination center.
 4. Manufacturers: Suppliers of vaccines for every vaccine centers.
 5. Developers: User who takes care of maintenance of the system/website.
- Clients are the users who use the website for booking their vaccine slots, getting certificate if vaccinated, doing online payment.
- Users who work at vaccination centers i.e., like doctors, nurse and other staff members who need access to update vaccination status of other users and check for validation of other user details like payment and criteria (age, disease) are classified as employees.
- Every vaccine center has a head person who handles all responsibilities like availability of vaccines and data for respective center and so website provides functions like making changes in vaccine slots availability, vaccines supply available at their center and so these users are categorized as managers.
- Manufacturers are the users who can update the stock/supply of vaccines at every vaccination center.
- Developers, these are the main users who hold responsibilities for managing entire website which includes key features like database and security, so they have access to every part of database. This category includes all technicians.

So, we can say that access to use website keeps on increasing from as we move from clients to employees to managers to developers. The website will provide a great security by ensuring that user within certain category cannot access other data. The most important user class is of Clients as they are in a huge number and so website should provide an environment which can be easy to use for this class, while the use of website will be more by managers, employees, and clients.

BACKGROUND READING:

➤ Description:

- In this section, we have collected the information from various published SRS and websites.
- Vaccines play an important role in human life. As we know India has a large population, it would be a difficult task to vaccinate the people in an efficient manner. So, we came up with this software to do it in an efficient and simpler way.
- In the existing systems the workflow is as mentioned above in workflow description part. It has features like:
 1. Finding nearest vaccine center by location, pin code and district wise.
 2. Registration is done by mobile number and Aadhar card both.
 3. Slot booking can be done when slots are available.
 4. After vaccination certificates can be downloaded which are for vaccination completion.
- Flaws in the existing systems are like:
 1. There is not a function through which online payment can be done.
 2. There is not proper management when problem of vaccine shortage is faced.
 3. Separate vaccines have separate website which means customer need to register at every website.
- To increase the efficiency of the vaccination drive, we must take care of the below-mentioned points:-
 - Availability of slot for vaccination
 - Vaccine center at every nearby location
 - Customer details and their vaccine details
 - Availability of several vaccines as per requirement
 - Different payment methods for ease of customers
 - Overall review and users-satisfaction by feedback
- Some better plans to boot the vaccination:-
 - By providing Door - Door service for possible vaccines.
 - By spreading awareness and camps for vaccination drives.

➤ References:

1. <https://www.cowin.gov.in>
2. <https://github.com/Corona-Vaccine-Management-System/Corona-Management>
3. <https://www.who.int/emergencies/diseases/novel-coronavirus-2019/covid-19-vaccines>
4. <https://www.academia.edu/28734143/HOSPITAL MANAGEMENT SYSTEM Software Requirement Specification>
5. <https://www.ukessays.com/essays/information-technology/definition-of-fact-finding-techniques-information-technology-essay.php>
6. <https://ukdiss.com/examples/hotel-management-system.php#citethis>

➤ Interviews and summary:

Interview 1:

Mock Interview Plan

Project Reference: SF/SJ/2021/10

System: Vaccination Drive

Interviewer: 1) Nakul Patel

Designation: Student at DAIICT

2) Shrey Mavani

Designation: Student at DAIICT

3) Divyesh Patel

Designation: Student at DAIICT

4) Soham Patel

Designation: Student at DAIICT

Interviewee: 1) Rakesh Kumar (**Role Play**)

Designation: Government Official for Health Care

Date: 4/10/2021

Time: 14:30

Duration: 40 minutes

Place: Google Meet (Online Mode)

Purpose of Interview:

- Preliminary meeting to identify problems and requirements for developing vaccination Software

Agenda:

- Discussion regarding the requirement for the website to make it User friendly.

Documents to be brought to the interview: Rough plan of Software

Interview 1 Summary:

Mock Interview Plan

Project Reference: SF/SJ/2021/10

System: Vaccination Drive

Interviewer: 1) Nakul Patel

Designation: Student at DAIICT

2) Shrey Mavani

Designation: Student at DAIICT

3) Divyesh Patel

Designation: Student at DAIICT

4) Soham Patel

Designation: Student at DAIICT

Interviewee: 1) Rakesh Kumar (**Role Play**) **Designation:** Government Official for Health Care

Date: 4/10/2021

Time: 14:30

Duration: 40 minutes

Place: Google Meet (Online Mode)

Summary of Interview:

1. Discussed the problem faced for organizing vaccine drives nowadays.
2. Discussed the feedbacks/complaints of citizens that they are facing during the vaccination.
3. Which are the necessary features required for registration at the vaccine center and discuss the different payment methods that they want in this system.
4. Discussion regarding choices given to clients to choose/locate their nearest vaccine center.
5. Deciding on how a website would distinguish/record the users, i.e., giving customer id to each user according to their category.

6. Discussion regarding the data to be shown on the website like number of vaccinated people, which can be further distinguished based on age, state, or district.

Interview2:

Mock Interview Plan

Project Reference: SF/SJ/2021/10

System: Vaccination Drive

Interviewer: 1) Nakul Patel

2) Shrey Mavani

Designation: Student at DAIICT

2) Shrey Mavani

Designation: Student at DAIICT

3) Divyesh Patel

Designation: Student at DAIICT

4) Soham Patel

Designation: Student at DAIICT

Interviewee: 1) Harika Jain (Role Play)

Designation: Doctor (Employee) at vaccine center

Date: 4/10/2021

Time: 16:30

Duration: 40 minutes

Place: Google Meet (Online Mode)

Purpose of Interview:

- Preliminary meeting to identify the problem related to vaccine availability during this pandemic and how they overcome the problem

Agenda:

- Problem with the lack of management
 - Follow-up actions in case of side effect of any vaccines

Documents to be brought to the interview: rough plan of Software

Interview 2 Summary:

Mock Interview Plan

Project Reference: SF/SJ/2021/10

System: Vaccination Drive

Interviewer: **1)** Nakul Patel

Designation: Student at DAIICT

2) Shrey Mavani

Designation: Student at DAIICT

3) Divyesh Patel

Designation: Student at DAIICT

4) Soham Patel

Designation: Student at DAIICT

Interviewee: **1)** Harika Jain (**Role Play**)

Designation: Doctor (Employee) at vaccine center

Date: 4/10/2021

Time: 16:30

Duration: 40 minutes

Place: Google Meet (Online Mode)

Summary of Interview:

1. Discussed the problem faced for organizing vaccine drive nowadays.
 2. Discussed the action taken in case of any side effects of any vaccines.
 3. List out the best vaccines available for diseases from other vaccines which are less efficient for that particular disease.
-

Interview 3:

Mock Interview Plan

Project Reference: SF/SJ/2021/10

System: Vaccination Drive

Interviewer: **1)** Nakul Patel

Designation: Student at DAIICT

2) Shrey Mavani

Designation: Student at DAIICT

3) Divyesh Patel

Designation: Student at DAIICT

4) Soham Patel

Designation: Student at DAIICT

Interviewee: **1)** Rajendra Mishra (**Role Play**)

Designation: Manufacturer

Date: 4/10/2021

Time: 20:00

Duration: 25 minutes

Place: Google Meet (Online Mode)

Purpose of Interview:

- Meeting to understand required functions from suppliers' view.

Agenda:

- Problem with the lack of management
- Follow-up action in case of shortage of vaccines

Documents to be brought to the interview: rough plan of Software

Interview 3 Summary:

Mock Interview Plan

Project Reference: SF/SJ/2021/10

System: Vaccination Drive

Interviewer: **1)** Nakul Patel

Designation: Student at DAIICT

2) Shrey Mavani

Designation: Student at DAIICT

3) Divyesh Patel

Designation: Student at DAIICT

4) Soham Patel

Designation: Student at DAIICT

Interviewee: **1)** Rajendra Mishra (**Role Play**)

Designation: Manufacturer

Date: 4/10/2021

Time: 20:00

Duration: 25 minutes

Place: Google Meet (Online Mode)

Summary of Interview:

1. Decided the functions needed to ease work of manufacturers.
 2. Decisions taken for resolving problems like shortage and unavailability of vaccines.
 3. Decided the necessary accesses for functions needed for manufacturers.
 4. Decided types, quantity and price of every vaccine that can be made available.
-

➤ Questionnaire:



Survey for vaccination website

This form is made to take survey from customers regarding vaccination drive.

 patelsb2002@gmail.com (not shared) Switch account 

* Required

Enter your full name. *

Your answer

1. From where you would prefer to take vaccine? *

- Private(hospitals) centers
 Government(aided) centers

2. What type of payment method would you use more preferably? *

- Online payment
- Net banking
- Cash/offline

3. Which identity would you prefer to use as login id? write your preferences *

	Aadhaar card/pan card	Phone Number	Email ID	Voter ID	Driving licence
First choice	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Second choice	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Third choice	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

4. Which of the given options would you prefer more for online vaccination registration? *

- Software/application
- Website
- Both

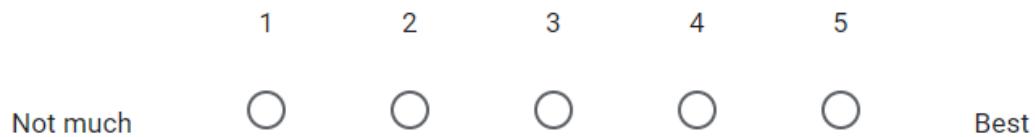
5. Which type of service would you prefer more? *

- Door-to-door
- At vaccine centers

6. which options do you prefer to find the nearest vaccine center?

- District
- Pincode
- Location

How much were you satisfied by the software? *



Feedback/Suggestions(if any) :

Your answer

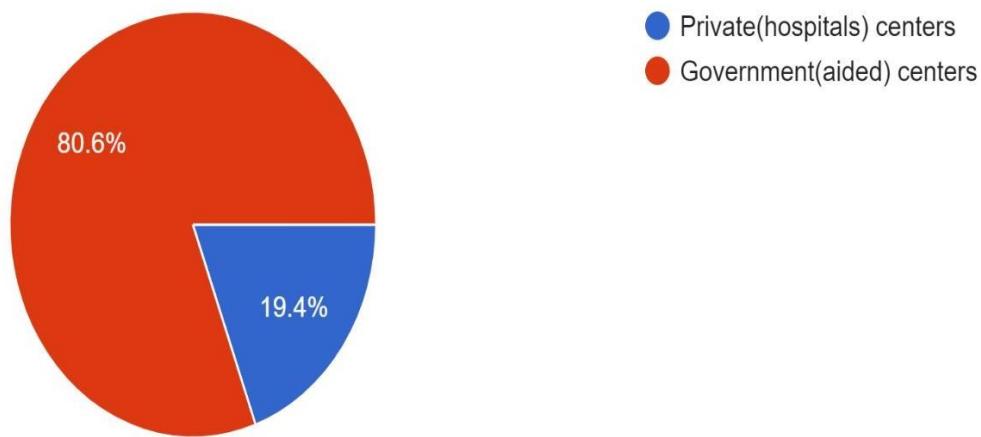
Submit

Clear form

➤ Responses:

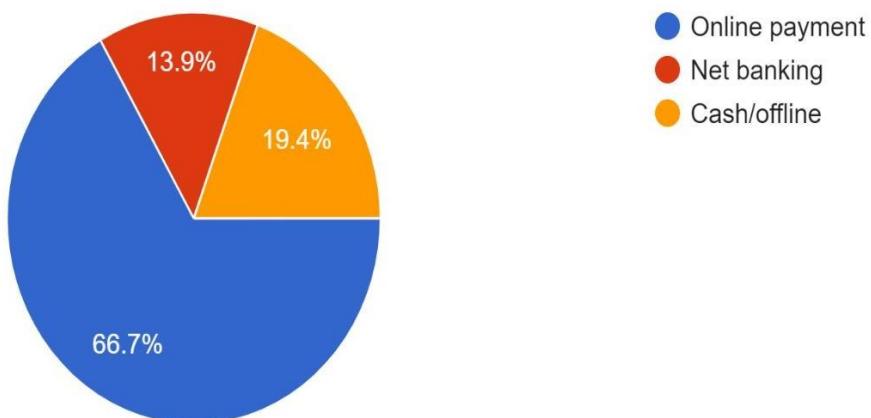
1. From where you would prefer to take vaccine?

36 responses

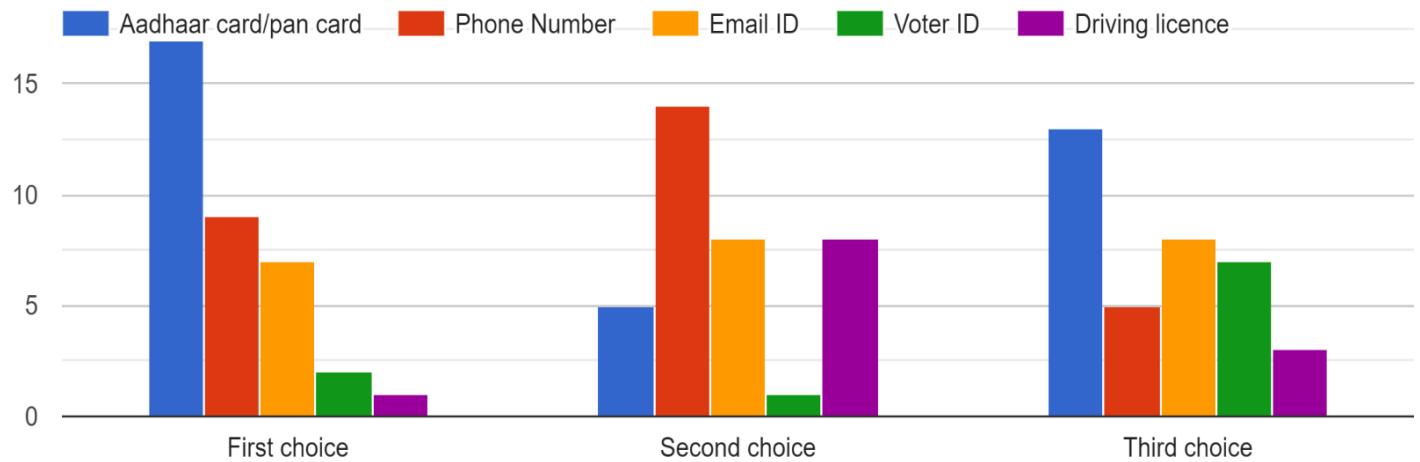


2. What type of payment method would you use more preferably?

36 responses

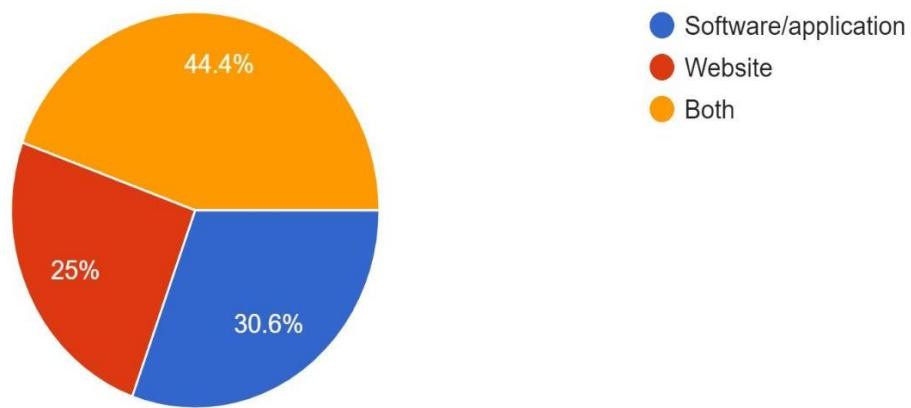


3. Which identity would you prefer to use as login id? write your preferences



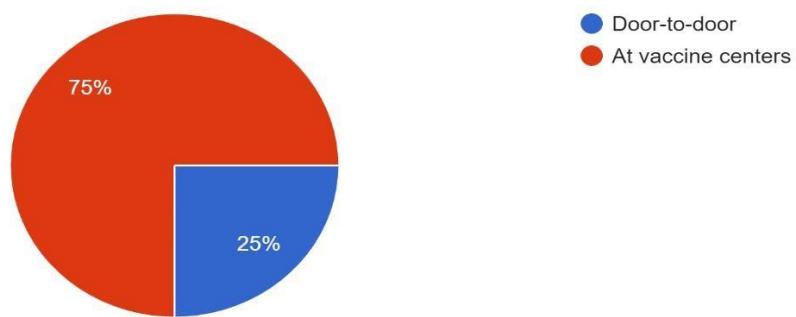
4. Which of the given options would you prefer more for online vaccination registration?

36 responses



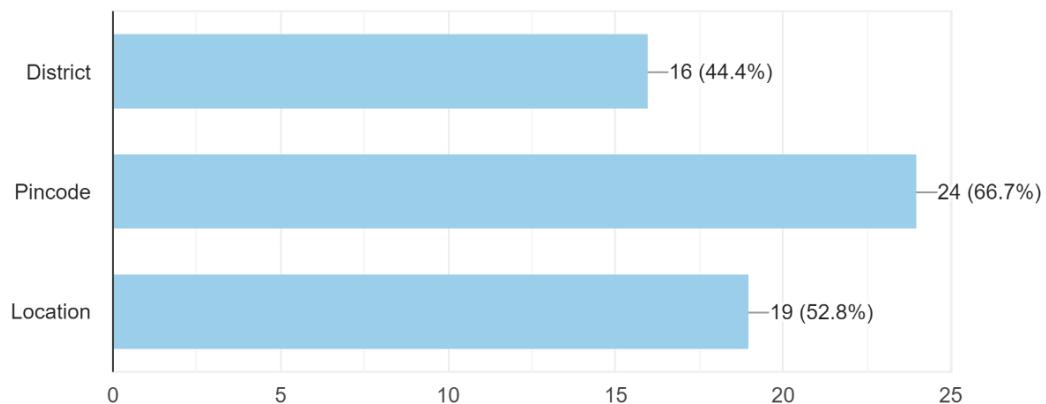
5. Which type of service would you prefer more?

36 responses



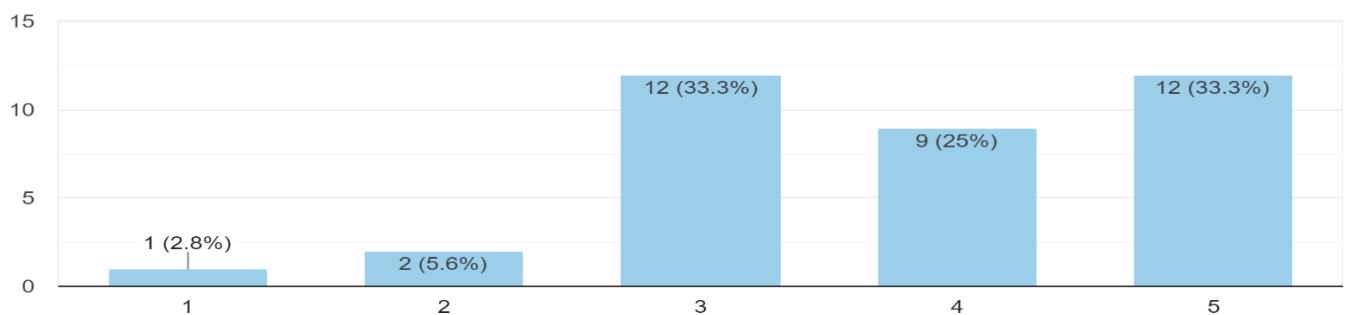
6. which options do you prefer to find the nearest vaccine center?

36 responses



How much were you satisfied by the software?

36 responses



➤ Observation:

System: Vaccination Drive Database

Project Reference: SF/SJ/2003/10

Observations by	1) Nakul Patel	Designation: Student at DAIICT
	2) Shrey Mavani	Designation: Student at DAIICT
	3) Divyesh Patel	Designation: Student at DAIICT
	4) Soham Patel	Designation: Student at DAIICT

Date: 8/10/2021 **Time:** 9:30

Duration: 50 minutes **Place:** Google Meet (Online mode)

Observations:

- Lack of vaccination centers more specifically government vaccination centers.
- Shortage of vaccines at the vaccine centers and even the supply of vaccines was not sufficient.
- Poor management at the vaccine centers due to less information regarding slots.
- Lack of services like online payment method.

Requirements:

- Contacting banks to approve online payment and net-banking.
- Need of more vaccine centres and more suppliers.
- Need for good database management, which can be easy to use, and can handle multiple traffic at a time.

- Fact Finding Chart:

Objective	Technique	Subject	Time
To get background on the vaccine drive.	Background reading	Company report and reading	0.5day
Discussion regarding the requirement for the website to make it User friendly.	Interview	Government official of healthcare	40 min
Follow-up actions in case of side effect of any vaccines.	Interview	Employee	40 min
Follow-up action in case of shortage of vaccines	Interview	Manufacturer	25 min
To get response from people regarding vaccination website	Questionnaire	Google form	1 day

LIST REQUIREMENTS

- Details of customers
 - List of all the users who have registered for vaccines and keep record of how many doses does he/she take.
- Details of vaccine centers
 - For anytime we need to know which vaccine centers are available for the vaccine that the user wants.
 - We must filter the centers on the base of the pin code that the user enters.
 - We must filter the centers based on time slots.

- Details of vaccines
 - List of all the vaccines with their price and their quantity.
- List of employees
 - List the user details according to their details and in which vaccine center they work under whom (manager).
- List of managers
 - List of managers of all the vaccine centers with their details.

OPERATING ENVIRONMENT :

➤ Hardware Interface Requirements

An efficient network,

- Hardware would consist of a WAN(Wide Area Network) with the use of Ethernet cable and Wireless networks having a high-Speed bandwidth which allows the high rate of data transfer which allows for the system to improve efficiency which in turn improves the backend of the business. It is advised to have a wired connection as the signals can be lost using a wireless network which can affect the performance of the system.

An efficient Laptop/desktop having the required details as:

1. The core process of intel i5/i7/i9
2. RAM 8 - 64 GB
3. SSD 512 GB - 1 TB.

Other requirements are: -Printer and a long-range Wi-Fi router.

➤ Software Interface Requirements

Software requirements for the database server:

Software	Requirements
Database Server	<ul style="list-style-type: none"> PostgreSQL MySQL Microsoft SQL Server
Operating System	<ul style="list-style-type: none"> Linux 5.7, 5.11, 6.0, or 6.6 Windows 7,8 or 10 Windows 2012 Standard Edition x86-64 Mac OS

The reason behind using this method is that if we XML files for building a database, it would have redundant data and which makes the database inefficient. Also, there are some advantages of the SQL database like merging different kinds of data and get required data from it. By using referential integrity, we can reduce the redundancy in the database and the most important thing is that we can also set the access integrity in the database.

The database will get data in 2 ways:

- 1) Online (through website/app)
- 2) Offline (at the vaccine center)

Software Requirement needed by the service providers:

Software	Requirements
Application Server	<ul style="list-style-type: none"> WebSphere Application Server J2EE WebLogic Glassfish
Java Development Kit	<ul style="list-style-type: none"> site24x7 Manage Engine Application Manager Java Performance Monitoring Junit NetBeans Apache Maven
Operating system	<ul style="list-style-type: none"> Linux 5.7, 5.11, 6.0, or 6.6 Windows 7,8 or 10 Mac OS

The data that will be coming in the system will assign a unique id/primary key by using which we can identify the data uniquely. This database will consist of each and every information of the vaccine center like an available stock of vaccine and slots information, customer details, and employee information, etc.

PRODUCT FUNCTIONS:

- We can divide all users of our system into four different types: user (customer), employee(staff), manager, manufacturer.

Customers	Employees	Managers	Manufacturers
Login/signup	Login	Login	Login
Registration	Update (user data and inventory)	Check and update inventory	Update inventory

- Customers:
 - Login/Sign Up:
 - ✓ Citizens can log in using Aadhar/Pan Card number or Phone number or Email ID or Driving license as their photo identity proof but as per the survey done using Googleform, most of the users like to login/sign up using Aadhar card number, so in our system, we used Aadhar Card number (default) to login/sign up their account (if the user is not having their Aadhar number handy, user can change their photo identity proof from options which include Driving License, PAN Card, Passport, etc.).

- Registration:
 - ✓ **Personal Information** – Users have to fill in their personal information which uniquely defines the user. This step involves the name of the person, age, gender, and the main part of this is photo identity proof.
 - ✓ **Vaccine Details** – After completion of the above step, the user will select the vaccine for which he/she wants to appear.
 - ✓ **Slot Booking** – As the user has selected the vaccine, now the user has to select the date and time at which the user is able to take the vaccine.
 - ✓ **Location** – This system provides three ways to locate their nearest vaccine centre using District, Pin code or using the user's location, but surveys say that most of the users want to locate it using their pin code.
 - ✓ **Payment** – Final step of registration payment. As per the survey, most users are familiar with online payment mode. But still our system provides payment using online mode, net banking or cash. Cash should be paid to the vaccine centre at the time of vaccination.
- Employees:
- Login:
 - ✓ Login using employee id assigned by management system.
- Update Data
 - ✓ **Verification**- One of the basic and important job of an employee is to verify the user using their photo identity proof and vaccinate the user if verification is successfully completed.
 - ✓ **Update User Data**- Employees have to update the user's data like the number of doses of a vaccine is completed.
 - ✓ **Update Inventory Data**- As one dose of a vaccine is used for the registered user; inventory data should be updated too as one dose of vaccine is used.
- Managers:
- Login
 - ✓ **Login**- Login using Manager id assigned by the management system.

- **Update Data**
 - ✓ **Inventory-** Manager will check the inventory of vaccines available in the inventory of the vaccine centre and if it is below the margin decided by the management, manager will order it from the manufacturer so that vaccine shortage wouldn't happen.

- Manufacturers:
- **Login**
 - ✓ Login using manufacturer id assigned by the management system.
- **Update Data-**
 - ✓ **Inventory-** Check the inventory of the manufacturer's vaccine, if it is below the margin decided by management, they can manufacture more of the same vaccine and if the manager of a specific vaccine center has ordered some vaccine the manufacturer will provide that vaccine to that vaccine center.

PRIVILEGES:

Functions	Users with access
Registration / login / sign-in	All users
Online Payment	Customers have access to their own Employees and managers have view access for all
Slots details	All users have view access Managers have editing access
Vaccine details	All users have view access Managers and manufacturers have edit access also

User database	Employees and managers have view and edit access for user database
employee database	Manager have view and edit access
Inventory	Managers and manufacturer have view and edit access both

ASSUMPTIONS:

- Users have devices that have internet service.
- Users have Aadhar cards associated with them so that we can uniquely identify them among other users.
- Users must have decent knowledge about how online registration works.
- All the vaccines are available in one inventory from where the manager orders the suppliers according to their need.
- All the vaccine centers have one and only one manager associated with it.

BUSINESS CONSTRAINTS:

- Always in touch with manufacturer to provide a more reliable vaccine supply so that there is no shortage of vaccines.
- Gathering enough suppliers so that stock of different type of vaccines can be obtained.
- Creation of digital infrastructure like getting good internet connection and fast accessible database server which can handle large amount of traffic at a time.
- Getting financial support from government and other NGO (trusts).

SECTION2: NOUN ANALYSIS

➤ Problem Description: -

- The database is to be designed keeping in mind the primary idea to make aneasy-to-use management system for all types of users.
- Every individual customer should be uniquely identified, to fulfill this requirement each of the customer will have to give his/her unique id number, which is Aadhar card as per survey, so to describe each customer uniquely we have their customer_id which is Aadhar card number. As, it has been taken as an assumption that each customer has Aadhar number which is unique for all. Alongside with Aadhar number customer have details like gender, first-name, middle-name, surname, birthdate, mobile number, and address. Address can be further divided in three sub-parts street, district, and state.
- There will also be data which contains details of employees working at the vaccine centers all around the country. Now, this employee details will have information like where the employee works (i.e., centre details), employee name, and work done by employee i.e., it will contain information about what type of work does employee do at the vaccination center which includes work categories like nurse, doctor, helper and even manager for each center. So, an employee can be manager, but each employee will work at only one center and each center will have exactly one manager. So, there will be only one manager at each center.
- Manager can also order different number of vaccines from manufactures according to the requirement and also keep track of the employee working to conduct those drives. Manager also have access to update and check the stocks of vaccine at vaccine center.
- Each of the vaccination center will be classified based on centre_id given to it which helps in uniquely identifying the centre. Other details like location of centre, centre name, and centre type. Centre type will describe the type of centre i.e., private or government. Vaccine center can also keep track of which and how many vaccines taken by customer.
- There will also be a data which contains all the details of vaccines. This data, Vaccine details will contain all the information about vaccines such as vaccine name, number of doses associated with that particular vaccine, vaccine price and vaccine_id which helps in uniquely identifying vaccines.

- Vaccines are needed so there is a need of data which have manufacturers details, this contains information like manufacturer_id and company_name. And manufacturers can also have access to change and update the inventory.
- The main data which plays a key role is inventory. This inventory database will contain crucial information's like number of vaccines supplied by each supplier. Even as it is known that each manufacturer can supply more than one type of vaccine, so alongside with company_name, manufacturer ID, the vaccine_id will also be mentioned in the inventory data. Now to keep track on number of vaccines supplied by each manufacturer available at each center another variable supplied_count will also be needed. This variable will help in allocating the number of slots and help in preventing the shortage of vaccines at vaccine centers. It can be updated whenever the manufacturer supplies vaccines to certain centre, subtraction from this count is done when slots are booked for that particular vaccine.
- Inventory database will also contain details like date and time of supply, type of company like private or government, cost of vaccine.
- Customers can book for particular time slot which are available for vaccination. Customers must be able to search their nearby vaccination center location.
- Manager also keep records of the people who are vaccinated and also generate the reports of the vaccination.
- In order to take vaccine each customer need to register and so we will keep all the registration details like customer_id (To identify customer), vaccine_id, vaccine_name, Dose_number, Center_id, date and time, payment method. In order to uniquely identify each registration, we will allocate unique registration ID.
- Employee have also access to view the registration details of all the customers so that they can verify it at the time of vaccination.
- There will be a data for slots which will contain information's like which type of vaccine will be available in that slot (vaccine_id), the place for which that slot is available (center_id), time and date for slot (slot_time, slot_date) and to uniquely identify each slot there will be a slot id associate

I. Noun (& Verb) Analysis.

1) Find the are nouns (entities) or verbs (relationships) in sentences of the problem description using Noun Analysis Method.

Nouns	Verbs
Location	Access
Management system	Allocate
Manager	Based
Manufacturer	Book
Manufacturer Details	Check
Manufacturer Id	Classify
Middle name	Contains
Mind	Described
Mobile number	Do
Number	Does
Nurse	Done
Order	Further
People	Generate
Private	Give
Register	Given
Registration details	Has
Report	Have
Requirement	Helps
Role	Identify
State	Include
Stock	Keep
Slot details	Manufactures
Street	Needed
Subtraction	Number
Supplied count	Plays
Supplier	Search
Vaccine Centre	Registers
Surname	Supplies
Time	Takes
Track	Update
Type	Will
Unique	Works
Users	
Vaccine ID	

Vaccine price	
Vaccine Name	
Vaccines	
Variable	
Work	
Customer ID	
Birthdate	
Inventory	
Centre	
Government	
Helper	
Database	
Doses	
Details	
District	
Conduct	
Employee	
Country	
Drives	
Idea	
Employee ID	
First Name	
Information	
Customer Details	
Doctor	
Category	
Gender	
key	
Address	
Access	
Cost	
Assumption	
Customer	
Company	
Company Name	
Data	
Last Name	
Date	
Employee details	
Slot details	
Slot time	
Slot date	
Slot id	
Age	

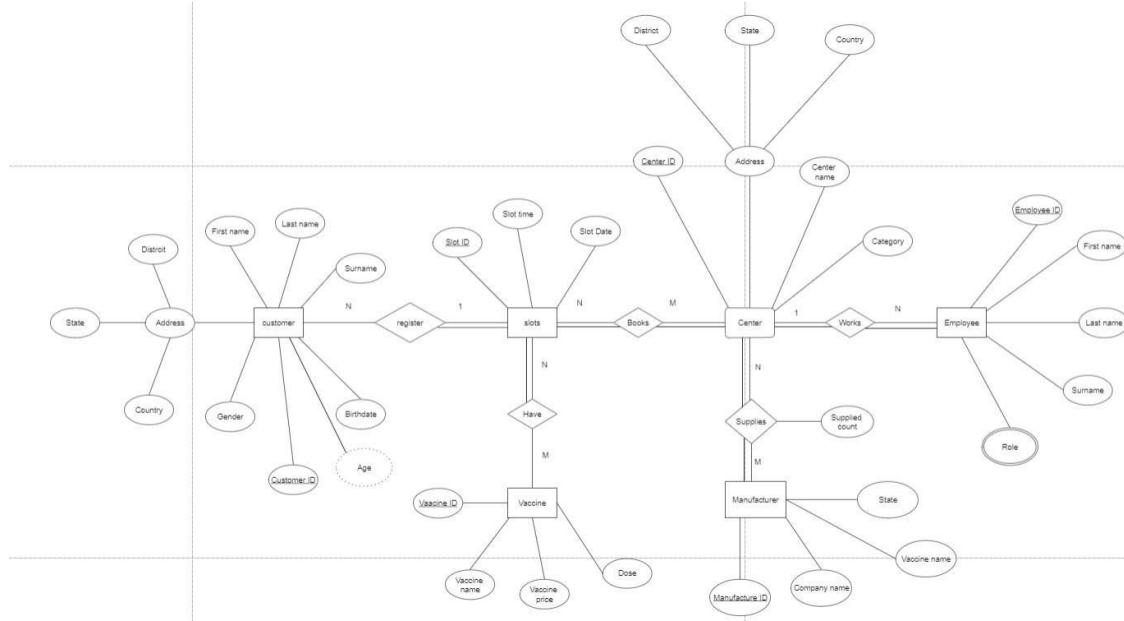
2) Criteria for Truncating Initial Noun List

Noun	Reject Reason
Date, Last name, Company name, Cost, Address, Gender, Category, First name, Employee ID, Country, District, Doses, Birthdate, Customer ID, Vaccinename, Vaccine price, Vaccine ID, Time, Surname, Supplied count, Street, State, Role, Mobile number, Middlename, Manufacture ID, Slot id, slot time, \$date, Age	Attributes
Company, Customer, Doctor, Government, Users, Type, Private, Order, Nurse, Manufacturer, Manager	General
Assumption, Access, Idea, Conduct, Variable, Track, Subtraction, Mind	Vague
Data, Drives, Database, Vaccine, Unique, Requirement, Management System	Irrelevant
Register, Allocate	Association
Information, Details, Helper, Centre, Work, Supplier, Stock, Report, People, Number, Location	Duplicate

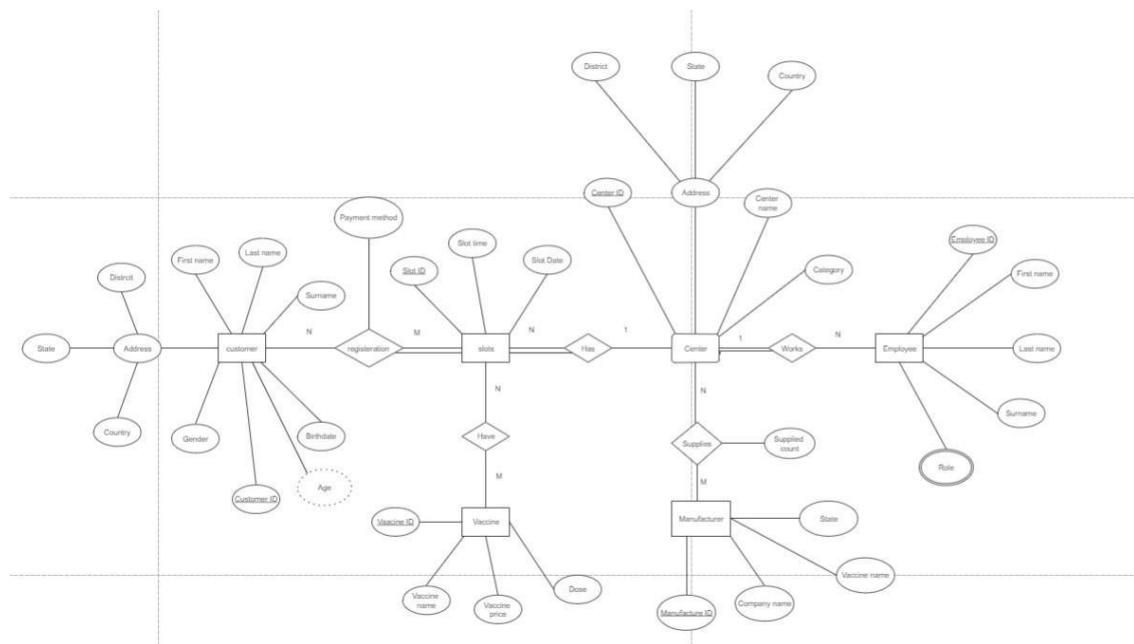
Candidate entity set	Candidate attribute set	Candidate relationship set
Customer details	Customer ID, First name, Last name, Surname, Birthdate, State, District, Country, gender, Age	Customers register slot
Manufacturer details	Manufacturer ID, Company name, state	Manufacturer supplies center
Vaccine Centre	Center ID, District, State, Country, center name, category	Vaccine center has slots
Vaccine details	Vaccine_id, vaccine name, vaccine price, doses	
Employee details	Employee ID, First name, Last name, Surname, Role	Employee works at center
Slot details	Slot id, slot time, slot date,	Slots have vaccine

SECTION3: ER-DIAGRAMS ALL VERSIONS

1) ERD VERSION 1

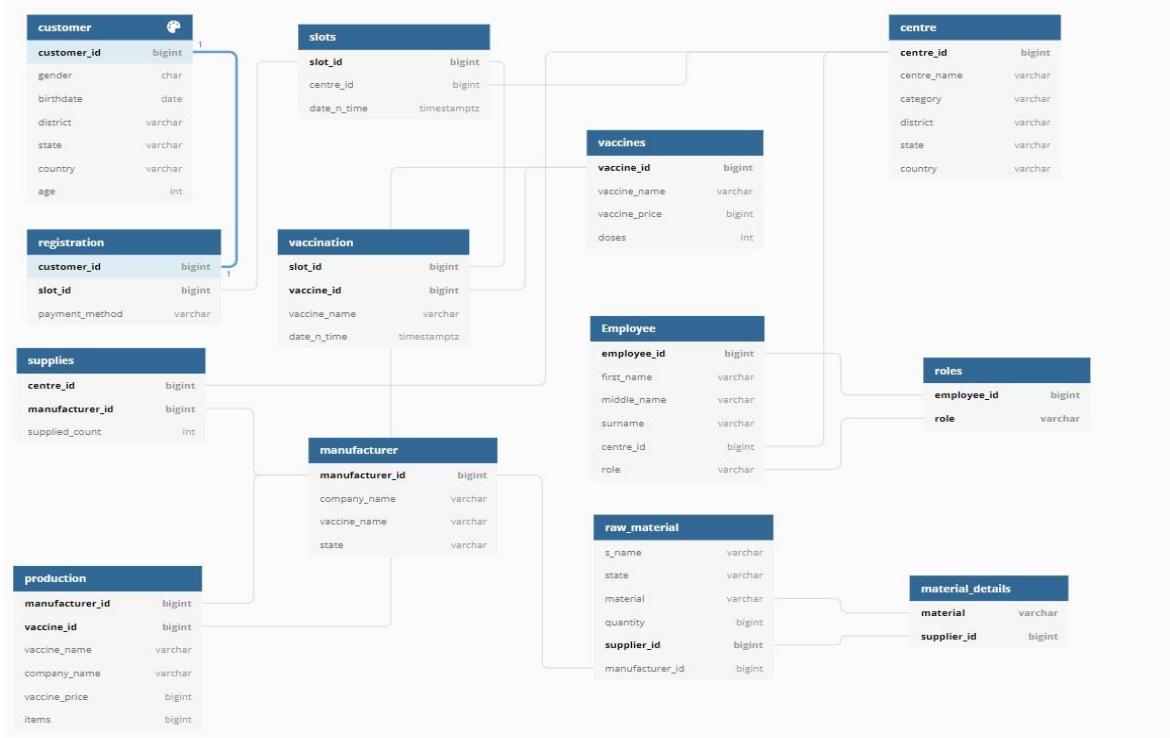


2) ERD VERSION 2



SECTION4: CONVERSION OF FINAL ER-DIAGRAM TO RELATIONAL MODEL

- **customers** (c_id, gender, birthdate, district, state, country, age).
- **centres** (centre_id, centre_name, category, district, state, country).
- **employee_details** (e_id, first_name, middle_name, surname, centre_id, role).
- **manufacturer** (manufacturer_id, company_name, vaccine_name, state).
- **supplies** (centre_id, manufacturer_id, supplied_count).
- **roles** (e_id, role).
- **slots** (slot_id, centre_id, date_n_time).
- **registration** (c_id, slot_id, payment_method).
- **vaccine** (vaccine_id, vaccine_name, vaccine_price, doses).
- **vaccination** (slot_id, vaccine_id, vaccine_name, date_n_time).
- **production** (manufacturer_id, vaccine_id, company_name, vaccine_name, vaccine_price, items)
- **raw_material** (s_id, s_name, manufacturer_id, state, material, quantity)
- **material_details** (material, s_id)



SECTION5: NORMALIZATION AND SCHEMA REFINEMENT

I. Normalization & Schema Refinement

1. List all the Relations & Schemas with all details (Original Design of Database)

- **customers** (c_id, gender, first_name, middle_name, surname birthdate, district, state, country, age).
- **centres** (centre_id, centre_name, category, district, state, country).
- **employee_details** (e_id, first_name, middle_name, surname, centre_id, role).
- **manufacturer** (manufacturer_id, company_name, vaccine_name, state).
- **supplies** (centre_id, manufacturer_id, supplied_count).
- **roles** (e_id, role).
- **slots** (slot_id, centre_id, date_n_time).
- **registration** (c_id, slot_id, payment_method).
- **vaccine** (vaccine_id, vaccine_name, vaccine_price, doses).
- **vaccination** (slot_id, vaccine_id, vaccine_name, date_n_time).
- **production** (manufacturer_id, vaccine_id, company_name, vaccine_name, vaccine_price, items)
- **raw_material** (s_id, s_name, manufacturer_id, state, quantity)
- **material_details** (material, s_id)

2. Identify and list all types of dependencies (PK, FK, Functional Dependencies) for each relation

Table	PK	FK
customers	c_id	
centres	centre_id	
employee_details	e_id	Centre_id
manufacturer	manufacturer_id	
supplies	centre_id, manufacturer_id	centre_id, manufacturer_id
roles	e_id, role	e_id
slots	slot_id	centre_id
registration	c_id, slot_id	c_id, slot_id
vaccine	vaccine_id	

vaccination	Slot_id, vaccine_id	Slot_id, vaccine_id
production	manufacturer_id, vaccine_id	manufacturer_id, vaccine_id
raw_material	S_id	manufacturer_id
material_details	Material, S_id	S_id

Functional Dependencies:

Table	Functional Dependencies
customers	Birthdate → age District → state, state → country C_id → gender, birthdate, district, state, country, age, first_name, middle_name, surname
centres	District → state, state → country Centre_id → centre_name, category, district, state, country Centre_name → centre_id, category, state
employee_details	E_id → first_name, middle_name, surname, centre_id, role
manufacturer	manufacturer_id → company_name, vaccine_name, state company_name, vaccine_name → state, manufacturer_id
supplies	centre_id, manufacturer_id → supplied_count
roles	
slots	slot_id, centre_id → date_n_time
registration	c_id, slot_id → payment_method
vaccine	vaccine_id → vaccine_name, vaccine_price, doses vaccine_name → vaccine_price, doses
vaccination	slot_id, vaccine_id → vaccine_name, date_n_time
production	manufacturer_id, vaccine_id → company_name, vaccine_name, vaccine_price, items company_name, vaccine_name → manufacturer_id, vaccine_id
raw_material	s_id → s_name, manufacturer_id, state, material, quantity s_name, manufacturer_id → s_id, state, material
material_details	

4) Normalize the database up to 1NF (scalar values)

- In the employee table, 'Role' is a multi-value attribute as One employee can have more than one role. So, a new table named 'Roles' is made which contains employee_id and role as attributes.

employee_details (e_id, first_name, middle_name, surname, centre_id)

roles (e_id, role)

- In the Raw_material table, 'material' is a multi-value attribute as One supplier can supply more than one material for given supplier_id. So, a new table named 'Material_details' is made which contains Supplier_id and material as attributes.

raw_material (s_id, s_name, manufacturer_id, state, quantity)

material_details (material, s_id)

- In the manufacturer table, 'Vaccine_name' is a multi-value attribute as a manufacturer can manufacture more than one vaccine. So, a new table named 'Manufacturing_details' is made which contains vaccine_name and manufacturer_id as attributes.

Manufacturer (manufacturer_id, company_name, vaccine_name, state).

Manufacturing_details (manufacturer_id, vaccine_name).

Candidate Keys: -

Customers: - C_id

Centres: - Centre_id and Centre_name

Employee_details: -E_id

Manufacturer: -manufacturer_id and {vaccine_name, company_name}

Supplies: - Centre_id, manufacturer_id

Slots: - Slot_id, centre_id

Registration: - c_id, slot_id

Vaccine: -Vaccine_id

Vaccination: - slot_id, vaccine_id

Production: - { manufacturer_id, vaccine_id } and company_name,
vaccine_name

Raw_material: - s_id and { s_name, manufacturer_id }

Manufacturing_details: - manufacturer_id

Table	Candidate Key	Prime attribute	Non-Prime attribute
Customers	C_id	C_id	gender, birthdate, district, state, country, age, first_name, middle_name, surname
Centres	Centre_id and Centre_name	Centre_id, Centre_name	category, district, state, country
Employee_details	E_id	E_id	first_name, middle_name, surname, centre_id, role

Manufacturer	manufacturer_id and {vaccine_name, company_name}	manufacturer_id, vaccine_name, company_name	state
Supplies	Centre_id, manufacturer_id	Centre_id, manufacturer_id	supplied_count
Slots	Slot_id, centre_id	Slot_id, centre_id	date_n_time
Registration	c_id, slot_id	c_id, slot_id	payment_method
vaccine	Vaccine_id	Vaccine_id	vaccine_name, vaccine_price, doses
vaccination	slot_id, vaccine_id	slot_id, vaccine_id	vaccine_name, date_n_time
production	{manufacturer_id, vaccine_id} and company_name, vaccine_name	manufacturer_id, vaccine_id, company_name, vaccine_name	vaccine_price, items
raw_material	s_id and {s_name, manufacturer_id}	s_id, s_name, manufacturer_id	state, material, quantity

5) Normalize the database further to 2NF (Remove Partial Dependencies)

Table	Reason
customers	In all the functional dependencies, we observed that the LHS of functional dependencies are not proper subset of candidate key (C_id) and RHS elements are all non-prime attributes.
centres	In this entity, all the functional dependencies are such that none of the LHS is proper subset of candidate key. So the table is in 2NF already.
employee_details	All the non-prime attributes are directly dependent on E_id so there is no partial dependency.

manufacturer	In all the functional dependencies, we observed that the LHS of functional dependencies are not proper subset of candidate key (manufacturer_id and {company_name, vaccine_name}).
supplies	LHS in the relational dependencies is the candidate key itself and not its proper subset so it is not partial dependent.
roles	
Slots	
Slots, Registration	LHS in the relational dependencies is the candidate key itself and not its proper subset so it is not partial dependent.
vaccine	In this entity, all the functional dependencies are such that none of the LHS is proper subset of candidate key. So the table is in 2NF already.
vaccination	LHS in the relational dependencies is the candidate key itself and not its proper subset so it is not partial dependent.
production	In all the functional dependencies, we observed that the LHS of functional dependencies are not proper subset of candidate key (manufacturer_id, vaccine_id and {company_name, vaccine_name}).
raw material	LHS in the relational dependencies is the candidate key itself and not its proper subset so it is not partial dependent.
material_details	

6) Identify (and document) List of redundancies exiting for the schema in 2NF

As we've mentioned above the reasons which proves that there are no redundancies in the tables.

7) Normalize it further to 3NF/BCNF (Remove Transitive Dependencies)

Condition – {LHS must be Candidate key or Super key} OR
 {RHS is a prime attribute}.

Table	Functional Dependencies
customers	Birthdate → age District → state, state → country C_id → gender, birthdate, district, state, country, age, first_name, middle_name, surname
centres	District → state, state → country Centre_id → centre_name, category, district, state, country Centre_name → centre_id, category, state
employee_details	In these functional dependencies, LHS is CK (E_id) and RHS is not prime but as first condition is satisfied so this table is 3NF.
manufacturer	In these functional dependencies, LHS is CK (manufacturer_id, {company_name, vaccine_name}) and RHS only have state attribute which is not prime but though LHS is satisfied so this table is in 3NF.
supplies	In these functional dependencies, LHS is CK (centre_id, manufacturer_id) and RHS (supplied_count) is not prime attribute but first condition is satisfied so this table is in 3NF.
Roles	
Slots	In these functional dependencies, LHS is CK (slot_id, centre_id) and RHS (date_n_time) is not prime attribute but first condition is satisfied so this table is in 3NF.
registration	In these functional dependencies, LHS is CK (c_id, slot_id) and RHS (payment_mehtod) is not prime attribute but first condition is satisfied so this table is in 3NF.

vaccine	vaccine_id → vaccine_name, vaccine_price, doses vaccine_name → vaccine_price, doses
vaccination	In these functional dependencies, LHS is CK (slot_id, vaccine_id) and RHS (vaccine_name, date_n_time) is not prime attribute but first condition is satisfied so this table is in 3NF.
production	In these functional dependencies, LHS is CK (manufacturer_id, vaccine_id, {company_name, vaccine_name}) and RHS (vaccine_price, items) is not prime attribute but first condition is satisfied so this table is in 3NF.
raw_material	In these functional dependencies, LHS is CK (manufacturer_id, vaccine_id, {s_id, s_name, manufacturer_id}) and RHS (state, material, quantity) is not prime attribute but first condition is satisfied so this table is in 3NF.
material_details	

In 3NF the functional dependencies of relation customers, centres, vaccine does not satisfy the condition for 3NF or their exist transitive relation in their functional dependencies.

The tables are decomposed in following form: -

- **Customers1** (c_id, gender, first_name, middle_name, surname)
- **Customers2** (c_id, birthdate, age)
- **Customers3** (c_id, district, state)
- **Customers4** (c_id, state, country)
- **Centres1** (centre_id, district, state)
- **Centres2** (centre_id, centre_name, category)
- **Centres3** (centre_id, country, state)

- **Vaccine1** (vaccine_id, vaccine_name)
- **Vaccine2** (vaccine_name, vaccine_price, doses)

Write down final relations with the schema.

- **Customers1** (c_id, first_name, middle_name, surname, gender)
- **Customers2** (c_id, birthdate, age)
- **Customers3** (c_id, district, state)
- **Customers4** (c_id, state, country)
- **Centres1** (centre_id, state, district)
- **Centres2** (centre_id, centre_name, category)
- **Centres3** (centre_id, country, state)
- **Vaccine1** (vaccine_id, vaccine_name)
- **Vaccine2** (vaccine_name, vaccine_price, doses)
- **employee_details** (e_id, first_name, middle_name, surname, centre_id)
- **manufacturer** (manufacturer_id, company_name, state)
- **manufacturer_vaccine** (manufacturer_id, vaccine_name)
- **supplies** (centre_id, manufacturer_id, supplied_count)
- **roles** (e_id, role).
- **slots** (slot_id, centre_id, date, time).
- **registration** (c_id, slot_id, payment_method).
- **vaccination** (slot_id, vaccine_id, vaccine_name, date, time).
- **production** (manufacturer_id, vaccine_id, company_name, vaccine_name, vaccine_price, items)
- **raw_material** (s_id, s_name, manufacturer_id, state, quantity)
- **material_details** (material, s_id)

**SECTION6: FINAL DDL
SCRIPTS, INSERT
STATEMENTS, 40 SQL QURIES
WITH SNAPSHOTS OF
OUTPUT OF EACH QUERY**

I. write DDL Scripts.

- 1) Recreate database by writing all Create Table statements (DDL) to accommodate the new design which is in 3NF/BCNF (removing your original version of relations)

```
create schema vaccine_drive1
set search path to "vaccine_drive1"
```

```
create table customers1 (
    c_id varchar (15),
    first_name varchar (30),
    middle_name varchar (30),
    surname varchar (30),
    gender char (1) not null,
    primary key (c_id),
    check (gender in ('M','F','T'))
);
```

```
create table customers2 (
    c_id varchar (15),
    birthdate date not null,
    age int,
    primary key (c_id)
);
```

```
create table customers3 (
    c_id varchar (15),
    district varchar (20) not null,
    state varchar (20),
    primary key (c_id)
);
```

```
create table customers4 (
    c_id varchar (15),
    state varchar (20),
    country varchar (30),
    primary key (c_id)
);
```

```
create table centres1 (
    centre_id bigint,
    state varchar (20) not null,
    district varchar (20),
    primary key (centre_id)
);
```

```
create table centres2 (
    centre_id bigint,
    centre_name varchar (30) not null,
    category varchar (10),
    check (category in ('private', 'government')),
    primary key (centre_id)
```

);

create table centres3 (

centre_id bigint,
country varchar (20),
state varchar (20) not null,
primary key (centre_id)
);

create table employee_details (

e_id bigint,
first_name varchar (30) not null,
middle_name varchar (30),
surname varchar (20) not null,
centre_id bigint,
primary key (e_id),
foreign key (centre_id) references
centres2(centre_id)
on delete cascade on update cascade
);

create table roles (

e_id bigint,
role varchar not null,
primary key (e_id, role),
foreign key (e_id) references
employee_details(e_id)

on delete cascade on update cascade

);

create table manufacturer (

manufacturer_id bigint,

company_name varchar (30) not null,

state varchar (20),

primary key(manufacturer_id)

);

create table manufacturer_vaccine (

manufacturer_id bigint,

vaccine_name varchar (20),

primary key (manufacturer_id, vaccine_name),

foreign key (manufacturer_id) references

manufacturer(manufacturer_id)

on delete cascade on update cascade

);

create table supplies (

centre_id bigint,

manufacturer_id bigint,

supplied_count int check(supplied_count>0),

primary key (centre_id, manufacturer_id),

foreign key (centre_id) references

centres2(centre_id)

on delete cascade on update cascade,

```
foreign key (manufacturer_id) references
manufacturer(manufacturer_id)
on delete cascade on update cascade
);
```

```
create table slots (
slot_id bigint,
centre_id bigint,
date date,
time varchar (20),
primary key (slot_id),
foreign key (centre_id) references
centres2(centre_id)
on delete cascade on update cascade
);
```

```
create table registration (
c_id varchar (15),
slot_id bigint,
payment_method varchar (20),
foreign key(c_id) references customers1(c_id)
primary key (c_id, slot_id),
on delete cascade on update cascade,
foreign key(slot_id) references slots(slot_id)
on delete cascade on update cascade,
check (payment_method in ('online', 'offline'))
```

);

```
create table vaccine1 (
    vaccine_id bigint,
    vaccine_name varchar (20),
    primary key (vaccine_id)
);
```

```
create table vaccine2 (
    vaccine_id bigint,
    vaccine_price bigint check(vaccine_price>0),
    doses int not null,
    primary key (vaccine_id)
);
```

```
create table vaccination (
    slot_id bigint,
    vaccine_id bigint,
    vaccine_name varchar (20) not null,
    date date,
    time varchar (20),
    primary key (slot_id, vaccine_id),
    foreign key(slot_id) references slots(slot_id)
        on delete cascade on update cascade,
    foreign key(vaccine_id) references
        vaccine1(vaccine_id)
        on delete cascade on update cascade
```

);

```
create Table production (
    manufacturer_id bigint,
    vaccine_id bigint,
    company_name varchar (30),
    vaccine_name varchar (30),
    vaccine_price bigint,
    items bigint,
    primary key (manufacturer_id, vaccine_id),
    foreign key(manufacturer_id) references
    manufacturer(manufacturer_id)
    on delete cascade on update cascade,
    foreign key(vaccine_id) references
    vaccine1(vaccine_id)
    on delete cascade on update cascade
);
```

```
create Table raw_material (
    s_id bigint,
    s_name varchar (30),
    manufacturer_id bigint,
    state varchar (30),
    quantity bigint,
    primary key(s_id),
    foreign key(manufacturer_id) references
    manufacturer(manufacturer_id)
```

on delete cascade on update cascade

);

```
create Table material_details (
    s_id bigint,
    material varchar (200),
    primary key (material, s_id),
    foreign key(s_id) references raw_material(s_id)
    on delete cascade on update cascade
);
```

Snapshots

Centre1: -

The screenshot shows the pgAdmin 4 interface. On the left, the 'Browser' pane displays a tree view of database objects under 'Tables (20)'. The 'centres1' table is selected. The 'Query Editor' pane contains the following SQL code:

```
1 SELECT * FROM vaccine_drive1.centres1
2 ORDER BY centre_id ASC
```

The results of the query are displayed in a table on the right:

centre_id	state	district
1	Bihar	Anaria
2	Goa	North Goa
3	Gujarat	Ahmedabad
4	Andhra Pradesh	Krishna
5	Maharashtra	Ahmednagar
6	Goa	South Goa
7	Anunachal Pradesh	Lohit
8	Andhra Pradesh	Kumool
9	Gujarat	Anand
10	Andhra Pradesh	Visakhapatnam
11	Anunachal Pradesh	kra daadi
12	Anunachal Pradesh	Tawang
13	Andhra Pradesh	Visakhapatnam
14	Gujarat	Bhavnagar
15	Arunachal Pradesh	Shiyomi
16	Bihar	Patna
17	Gujarat	Banaskantha
18	Anunachal Pradesh	Tirap
19	Maharashtra	Mumbai City
20	Maharashtra	Nashik

Centre2: -

The screenshot shows the pgAdmin 4 interface with the database '201901043_db' selected. The left sidebar lists tables, and the right pane displays the results of a query:

```
1 SELECT * FROM vaccine_drive1.centres2
2 ORDER BY centre_id ASC
```

centre_id	centre_name	category
1	Delight Sun Clinic	Government
2	Another life Hospital	Government
3	West valley hospital	Private
4	Willow Green Hospital	Government
5	Hopewell Hospital	Government
6	Community Health Service	Private
7	Medica Zone	Private
8	Family Wellness Center	Government
9	Injury Care Medics	Government
10	Ablecare	Private
11	Blue Line Health Care Co.	Government
12	Ultra Care House	Government
13	Ach hospital	Government
14	Redstar Hospital	Government
15	Flowerflake Clinic	Government
16	lifeflash Clinic	Private
17	Winter warm	Government
18	Angelwalk Hospital	Private
19	Intense Hospital	Government
20	Summer spring Hospital	Private

Centres3: -

The screenshot shows the pgAdmin 4 interface with the database '201901043_db' selected. The left sidebar lists tables, and the right pane displays the results of a query:

```
1 SELECT * FROM vaccine_drive1.centres3
2 ORDER BY centre_id ASC
```

centre_id	country	state
1	India	Bihar
2	India	Goa
3	India	Gujarat
4	India	Andhra Pradesh
5	India	Maharashtra
6	India	Goa
7	India	Arunachal Pradesh
8	India	Andhra Pradesh
9	India	Gujarat
10	India	Andhra Pradesh
11	India	Arunachal Pradesh
12	India	Arunachal Pradesh
13	India	Andhra Pradesh
14	India	Gujarat
15	India	Arunchal Pradesh
16	India	Bihar
17	India	Gujarat
18	India	Arunchal Pradesh
19	India	Maharashtra
20	India	Maharashtra

Customers1: -

pgAdmin 4

File Object Tools Help

Browser

- > Foreign Tables
- > Functions
- > Materialized Views
- > Procedures
- > Sequences
- > Tables (20)
 - > centres1
 - > centres2
 - > centres3
 - > customers1
 - > customers2
 - > customers3
 - > customers4
 - > employee_details
 - > manufacturer
 - > manufacturer_vaccine
 - > material_details
 - > production
 - > raw_material
 - > registration
 - > roles
 - > slots
 - > supplies
 - > vaccination
 - > vaccine1
 - > vaccine2
- > Trigger Functions
- > Types
- > Views
- > Subscriptions
- > postgres
- > Login/Group Roles
- > Tablespaces

201901043_db/postgres@PostgreSQL 13

Query Editor Query History

	c_id	first_name	middle_name	surname	gender	
1	59	Shanta	Subrahmany	Jain	F	
2	60	Manes	Anand	Rao	M	
	61	Shreya	Nishant	Ahmed	F	
	62	Prashant	Devaraj	Nilbanupudi	M	
	63	Narinder	Nand	Joshi	M	
	64	Rakhi	Balwinder	Jain	F	
	65	Shreesth	Rajneesh	Ahmad	M	
	66	Kumari	Rajiv	D'Cruze	F	
	67	Kanta	Venkat	Kaur	F	
	68	Avinash	Amardeep	Choudhary	M	
	69	Mitul	Prasad	Misra	T	
	70	Gobind	Chandrankant	Sharma	M	
	71	Gayatri	Sree	Singh	F	
	72	Ranq	Sekhar	Tamboli	M	
	73	Pallav	Narayanan	Das	M	
	74	Aravind	Vivek	Bachchan	M	
	75	Nikhil	Ram	Chaudhary	M	
	76	960238912797	Sunil	Sankar	Sharma	M
	77	97508237916	Chander	Sarvesh	Ahmed	F
	78	977411460866	Rajeev	Ratna	Gadhwai	M
	79	982915072029	Indu	Jai	Jain	F
	80	997979339316	Jagjit	Dharma	Jain	M

Customers2: -

pgAdmin 4

File Object Tools Help

Browser

- > Foreign Tables
- > Functions
- > Materialized Views
- > Procedures
- > Sequences
- > Tables (20)
 - > centres1
 - > centres2
 - > centres3
 - > customers1
 - > customers2
 - > customers3
 - > customers4
 - > employee_details
 - > manufacturer
 - > manufacturer_vaccine
 - > material_details
 - > production
 - > raw_material
 - > registration
 - > roles
 - > slots
 - > supplies
 - > vaccination
 - > vaccine1
 - > vaccine2
- > Trigger Functions
- > Types
- > Views
- > Subscriptions
- > postgres
- > Login/Group Roles
- > Tablespaces

201901043_db/postgres@PostgreSQL 13

Query Editor Query History

	c_id	birthdate	date	age
1	59	1954-04-15		67
2	60	2001-07-11		20
	61	1958-01-21		63
	62	1986-11-29		34
	63	2000-11-30		20
	64	1996-08-21		25
	65	1965-03-26		56
	66	1998-05-13		23
	67	1973-04-21		48
	68	1977-08-23		44
	69	1960-03-09		61
	70	1984-05-21		37
	71	1968-10-22		53
	72	1961-05-13		60
	73	1970-04-19		51
	74	1977-08-12		43
	75	1980-04-04		41
	76	1989-04-11		32
	77	1972-04-21		49
	78	1967-02-02		54
	79	1988-06-02		33
	80	1954-04-19		67

Customers3: -

The screenshot shows the pgAdmin 4 interface with the following details:

- Browser:** Shows the database structure with the `customers3` table selected.
- Query Editor:** Contains the SQL query:

```
1 SELECT * FROM vaccine_drive1.customers3
2 ORDER BY c_id ASC
```
- Data Output:** Displays the results of the query as a table with columns: `c_id`, `district`, `state`. The data includes 80 rows of Indian state information.

c_id	district	state
59	Bhiwani	Haryana
60	West District	Sikkim
61	Zunheboto	Nagaland
62	Malappuram	Kerala
63	Tawang	Arunachal Pradesh
64	Kiphire	Nagaland
65	Salem	Tamil Nadu
66	West District	Sikkim
67	Dimapur	Nagaland
68	Nagaon	Assam
69	Giridih	Jharkhand
70	Mulugu	Telangana
71	West Khasi Hills	Meghalaya
72	Churachandpur	Manipur
73	Mewat	Haryana
74	West District	Sikkim
75	Amritsar	Punjab
76	Satna	Madhya Pradesh
77	Mehsana	Gujarat
78	Indore	Madhya Pradesh
79	Darjeeling	West Bengal
80	Gulbarga	Karnataka

Customers4: -

The screenshot shows the pgAdmin 4 interface with the following details:

- Browser:** Shows the database structure with the `customers4` table selected.
- Query Editor:** Contains the SQL query:

```
1 SELECT * FROM vaccine_drive1.customers4
2 ORDER BY c_id ASC
```
- Data Output:** Displays the results of the query as a table with columns: `c_id`, `state`, `country`. The data includes 80 rows of Indian state information.

c_id	state	country
59	Haryana	India
60	Sikkim	India
61	Nagaland	India
62	Kerala	India
63	Arunachal Pradesh	India
64	Nagaland	India
65	Tamil Nadu	India
66	Sikkim	India
67	Nagaland	India
68	Assam	India
69	Jharkhand	India
70	Telangana	India
71	Meghalaya	India
72	Manipur	India
73	Haryana	India
74	Sikkim	India
75	Punjab	India
76	Madhya Pradesh	India
77	Gujarat	India
78	Madhya Pradesh	India
79	West Bengal	India
80	Karnataka	India

Employee_details:-

The screenshot shows the pgAdmin 4 interface with the 'employee_details' table selected in the left sidebar. The main area displays the following SQL query and its results:

```
1 SELECT * FROM vaccine_drive1.employee_details
2 ORDER BY e_id ASC
```

e_id	first_name	middle_name	surname	centre_id
59	Sunita	Rashmi	Narang	8
60	Vijaya	Nihal	Narang	5
61	Kamini	Sanjay	D'Cruz	7
62	Vimal	Kalyan	Sharma	14
63	Anupama	Sanjita	Rao	2
64	Sanjay	Dilip	Sharma	5
65	Nanda	Puneet	Chaudhri	12
66	Deo	Sanjay	D'Cruz	5
67	Rohan	Kalyan	Sharma	10
68	Devika	Sanjita	Rao	4
69	Kaur	Dilip	Sharma	16
70	Purushottam	Puneet	Chaudhri	19
71	Jayant	Harish	Ahmed	2
72	Dhananjay	Indrajit	Patel	9
73	Dipti	Anant	Rao	11
74	Kalyana	Laxman	Patel	8
75	Chand	Ajay	Joshi	17
76	Jayant	Kuldeep	Bachchan	8
77	Dhananjay	Navdeep	D'Cruze	16
78	Dipti	Kumaran	Anand	17
79	Kalyana	Mukul	Chaudhri	17
80	Chand	Inderjit	Ahmed	19

Manufacturer: -

The screenshot shows the pgAdmin 4 interface with the 'manufacturer' table selected in the left sidebar. The main area displays the following SQL query and its results:

```
1 SELECT * FROM vaccine_drive1.manufacturer
2 ORDER BY manufacturer_id ASC
```

manufacturer_id	company_name	state
1	Bharat biotech International Ltd	Telangana
2	Cadila Healthcare Limited	Gujarat
3	Cadila Pharmaceuticals Ltd	Gujarat
4	Chiron Behering	Gujarat
5	Dano Vaccine & Biological Pvt. Ltd.	Telangana
6	Green signal BioPharma Ltd	Tamil Nadu
7	Panacea	Himachal Pradesh
8	Rambaxy Lab	Karnataka
9	Serum Institute of India	Maharashtra
10	Shanth Biotechnics Ltd	Andhra Pradesh
11	GSK Asia Pvt. Ltd.	Maharashtra
12	Sanofi Pasteur India Pvt Ltd	Maharashtra
13	CRI	Himachal Pradesh
14	Pasteur Institute of India	Tamil Nadu
15	Human biological Institute	Telangana
16	Halfkine, AcharyaDondeMarg	Maharashtra

manufacturer_vaccine: -

The screenshot shows the pgAdmin 4 interface with the 'manufacturer_vaccine' table selected in the left sidebar under the 'Tables' section. The main pane displays the following SQL query and its results:

```
1 SELECT * FROM vaccine_drive1.manufacturer_vaccine
2 ORDER BY manufacturer_id ASC, vaccine_name ASC
```

	manufacturer_id	vaccine_name
2	1	Rotavirus
3	1	Typhoid
4	2	Rabies vaccine
5	3	H1N1 Vaccine
6	4	Rabies vaccine
7	5	Rotavirus
8	5	TT
9	6	HPV
10	7	H1N1
11	8	Rabies
12	8	Typhoid
13	9	BCG
14	9	TT
15	10	Hep B Bulk
16	11	H1N1
17	12	TT
18	12	Typhoid
19	13	Hep B Bulk
20	14	HPV
21	15	Rabies
22	15	TT
23	16	Corona Vaccine

material_details: -

The screenshot shows the pgAdmin 4 interface with the 'material_details' table selected in the left sidebar under the 'Tables' section. The main pane displays the following SQL query and its results:

```
1 SELECT * FROM vaccine_drive1.material_details
2 ORDER BY s_id ASC, material ASC
```

	s_id	material
30	20	Sorbito
31	21	Magnesium chloride hexahydrate
32	22	Neomycin
33	22	Sorbito
34	23	Potassium chloride (in bulk)
35	24	Formaldehyde
36	24	Thimerosal
37	25	Neomycin
38	26	Erlenmeyer flasks, of fused quartz or other fused silica
39	27	Emulsifiers
40	27	Hydrochloric acid
41	28	Yeast Protein
42	29	Cholesterol
43	29	Hydrochloric acid
44	30	Hydrochloric acid
45	31	Gelatin
46	31	Plastic bag with inbuilt filter for the sterile filtration, storage and transfer of biopharmaceuti
47	32	Gelatin
48	33	Monobasic potassium phosphate
49	34	Thimerosal
50	35	Liquid storage bags, of polymers of ethylene
51	36	Erlenmeyer flasks, of other glass having a linear coefficient of expansion not exceeding 5 x

Production: -

The screenshot shows the pgAdmin 4 interface with the 'production' table selected in the left sidebar. The main area displays the results of the following SQL query:

```

1 SELECT * FROM vaccine_drive1.production
2 ORDER BY manufacturer_id ASC, vaccine_id ASC
  
```

The results are presented in a grid table with the following columns and data:

manufacturer_id	vaccine_id	company_name	vaccine_name	vaccine_price	items
2	1	Bharat biotech International Ltd	Corona Vaccine	650	5986
3	1	Bharat biotech International Ltd	Rotavirus	800	3852
4	2	Cadila Healthcare Limited	Rabies vaccine	300	2200
5	3	Cadila Pharmaceuticals Ltd	H1N1 Vaccine	175	2797
6	4	Chiron Behring	Rabies vaccine	275	5005
7	5	Dano Vaccine & Biological Pvt. Ltd.	TT	5	573
8	5	Dano Vaccine & Biological Pvt. Ltd.	Rotavirus	700	9535
9	6	Green signal BioPharma Ltd	HPV	2500	7341
10	7	Panacea	H1N1	150	5869
11	8	Ranbaxy Lab	Typhoid	1200	2228
12	8	Ranbaxy Lab	Rabies	250	7670
13	9	Serum Institute of India	TT	5	4269
14	9	Serum Institute of India	BCG	625	2596
15	10	Shantha Biotechnics Ltd	Hep B Bulk	110	4003
16	11	GSK Asia Pvt. Ltd	H1N1	200	3956
17	12	Sanofi Pasteur India Pvt Ltd	Typhoid	1475	3528
18	12	Sanofi Pasteur India Pvt Ltd	TT	5	3789
19	13	CRI	Hep B Bulk	120	4625
20	14	Pasteur institute of India	HPV	2600	3427
21	15	Human biological institute	Rabies	215	5787
22	15	Human biological institute	TT	5	8837
23	16	Haffkine, AcharyaDondeMarg	Corona Vaccine	500	9887

Raw_material: -

The screenshot shows the pgAdmin 4 interface with the 'raw_material' table selected in the left sidebar. The main area displays the results of the following SQL query:

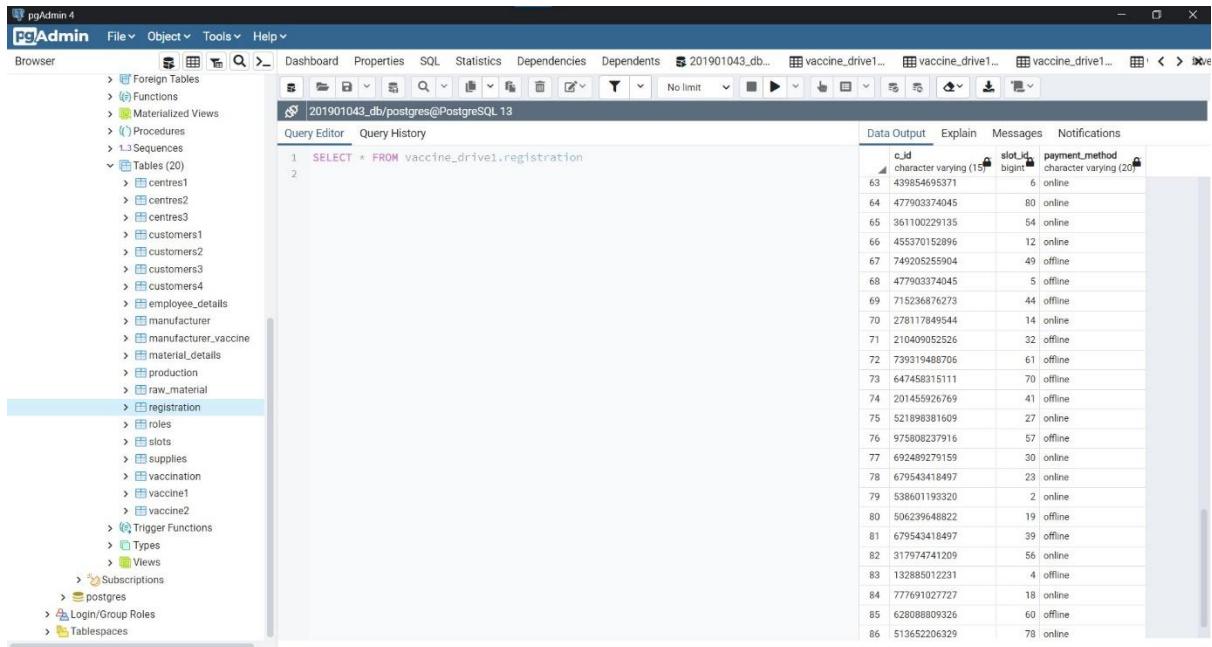
```

1 SELECT * FROM vaccine_drive1.raw_material
2 ORDER BY s_id ASC
  
```

The results are presented in a grid table with the following columns and data:

s_id	s_name	manufacturer_id	state	quantity
15	Triveni Interchem Pvt. Ltd	3	Delhi	10000
16	Forbes Pharmaceutical	2	Tamil Nadu	18000
17	Khusi Chemicals Pvt Ltd	1	Maharashtra	42000
18	AVA Chemicals Private Limited	2	Gujarat	20000
19	Vet Bio Chem Pvt Ltd	3	Rajasthan	35000
20	Ultramarines India (P) Ltd	4	Maharashtra	23000
21	K-Tech (India) Limited	5	Rajasthan	55000
22	Penta Bioscience Products	6	Gujarat	12000
23	Chemicals & Associates, New Delhi	7	Tamil Nadu	4000
24	Vats International	8	Maharashtra	9500
25	Akash Purochem Private Limited	9	Gujarat	95500
26	Yashica Pharmaceuticals Private Limited	10	Tamil Nadu	85000
27	Prasid Chemicals Limited	11	Delhi	13000
28	Excel Metal & Engg Industries	12	Rajasthan	11000
29	Triveni Aromatics And Perfumery Private Limited	13	Karnataka	120000
30	Yogi Dye Chem Industries	14	Delhi	30000
31	Ganesh Benzoplast Limited	15	Tamil Nadu	50000
32	BioBaxy Technologies India	16	Maharashtra	15000
33	Cygnus Healthcare Specialties Private Limited	9	Gujarat	25000
34	Suboney Chemicals Pharmaceuticals (P) Limited	8	Rajasthan	30000
35	Healthy Life Pharma Private Limited	7	Maharashtra	10000
36	Bioprex Labs	6	Rajasthan	35000

registration: -

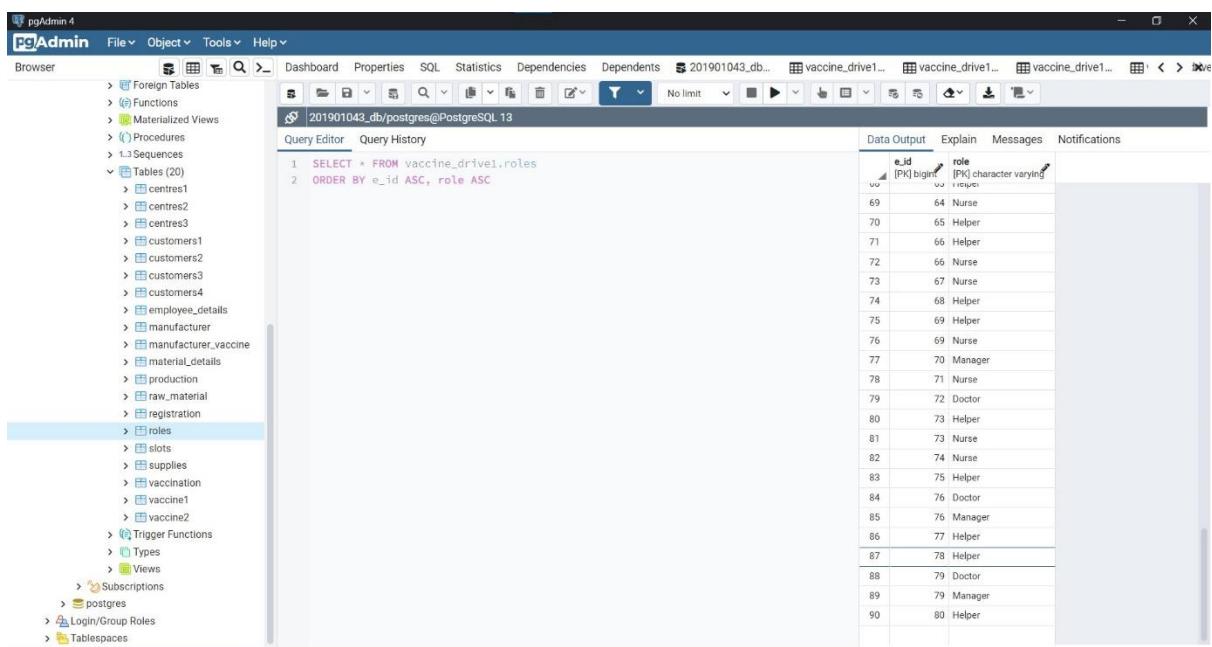


The screenshot shows the pgAdmin 4 interface with the 'registration' table selected in the left-hand tree view. The main pane displays the results of a SELECT query:

```
1 SELECT * FROM vaccine_drive1.registration
2
```

c_id	slot_id	payment_method
character varying (15)	bigint	character varying (20)
63	439854695371	6
64	477903374045	80
65	36110029135	54
66	455370152896	12
67	749205255904	49
68	477903374045	5
69	715236876273	44
70	278117849544	14
71	210409052526	32
72	739319488706	61
73	647458315111	70
74	201465926769	41
75	521898381609	27
76	975808237916	57
77	692489279159	30
78	679543418497	23
79	538601193320	2
80	506239648822	19
81	679543418497	39
82	317974741209	56
83	132885012231	4
84	777691027727	18
85	628088809326	60
86	513652206329	78
87		online

roles: -



The screenshot shows the pgAdmin 4 interface with the 'roles' table selected in the left-hand tree view. The main pane displays the results of a SELECT query:

```
1 SELECT * FROM vaccine_drive1.roles
2 ORDER BY e_id ASC, role ASC
```

e_id	role
[PK] bigint	[PK] character varying
69	64 Nurse
70	65 Helper
71	66 Helper
72	66 Nurse
73	67 Nurse
74	68 Helper
75	69 Helper
76	69 Nurse
77	70 Manager
78	71 Nurse
79	72 Doctor
80	73 Helper
81	73 Nurse
82	74 Nurse
83	75 Helper
84	76 Doctor
85	76 Manager
86	77 Helper
87	78 Helper
88	79 Doctor
89	79 Manager
90	80 Helper

slots: -

The screenshot shows the pgAdmin 4 interface with the 'slots' table selected in the left sidebar. The main area displays the following SQL query and its results:

```
1 SELECT * FROM vaccine_drive1.slots
2 ORDER BY slot_id ASC
```

slot_id	centre_id	date	time
59	59	15	2022-03-27 14:00 - 16:00
60	60	15	2022-11-15 16:00 - 17:00
61	61	16	2022-12-03 8:00 - 10:00
62	62	16	2022-06-19 10:00 - 12:00
63	63	16	2022-01-17 14:00 - 16:00
64	64	16	2022-09-24 16:00 - 17:00
65	65	17	2022-08-20 8:00 - 10:00
66	66	17	2022-06-14 10:00 - 12:00
67	67	17	2022-05-13 14:00 - 16:00
68	68	17	2022-06-04 16:00 - 17:00
69	69	18	2022-02-05 8:00 - 10:00
70	70	18	2022-07-19 10:00 - 12:00
71	71	18	2022-04-15 14:00 - 16:00
72	72	18	2022-12-11 16:00 - 17:00
73	73	19	2022-11-06 8:00 - 10:00
74	74	19	2022-03-31 10:00 - 12:00
75	75	19	2022-06-27 14:00 - 16:00
76	76	19	2022-06-03 16:00 - 17:00
77	77	20	2022-07-11 8:00 - 10:00
78	78	20	2022-10-19 10:00 - 12:00
79	79	20	2022-01-23 14:00 - 16:00
80	80	20	2022-05-28 16:00 - 17:00

Supplies: -

The screenshot shows the pgAdmin 4 interface with the 'supplies' table selected in the left sidebar. The main area displays the following SQL query and its results:

```
1 SELECT * FROM vaccine_drive1.supplies
2 ORDER BY centre_id ASC, manufacturer_id ASC
```

centre_id	manufacturer_id	supplied_count
11	6	6
12	6	12
13	7	4
14	7	10
15	8	1
16	8	8
17	9	3
18	9	11
19	9	13
20	10	2
21	10	9
22	11	3
23	11	6
24	12	14
25	13	4
26	13	5
27	13	7
28	14	4
29	15	4
30	15	6
31	15	7
32	15	8

vaccination: -

The screenshot shows the pgAdmin 4 interface with the following details:

- Left Panel (Browser):** Shows the database structure with the "vaccination" table selected.
- Query Editor:** Contains the following SQL query:

```
1 SELECT * FROM vaccine_drive1.vaccination
2 ORDER BY slot_id ASC, vaccine_id ASC
```
- Data Output:** Displays the results of the query in a table format. The columns are: slot_id, vaccine_id, vaccine_name, date, and time. The data includes various vaccinations like Rotavirus, Hep B Bulk, Corona, Rabies, Typhoid, TT, H1N1, HPV, Hep B Bulk, BCG, and Rotavirus across different dates and times.

Vaccine1: -

The screenshot shows the pgAdmin 4 interface with the following details:

- Left Panel (Browser):** Shows the database structure with the "vaccine1" table selected.
- Query Editor:** Contains the following SQL query:

```
1 SELECT * FROM vaccine_drive1.vaccine1
2 ORDER BY vaccine_id ASC
```
- Data Output:** Displays the results of the query in a table format. The columns are: vaccine_id and vaccine_name. The data includes Typhoid, Corona, H1N1, Rabies, TT, HPV, Hep B Bulk, BCG, and Rotavirus.
- Status Bar:** Shows a green message: "Successfully run. Total query runtime: 222 msec. 9 rows affected."

vaccine2: -

The screenshot shows the pgAdmin 4 interface. On the left, the 'Browser' pane lists various database objects: Foreign Tables, Functions, Materialized Views, Procedures, Sequences, Tables (20), Triggers, Types, Views, Subscriptions, and PostgreSQL system objects. The 'vaccine2' table is selected and highlighted with a blue background. In the center, the 'Query Editor' tab is active, displaying the following SQL query:

```
1 SELECT * FROM vaccine_drive1.vaccine2
2 ORDER BY vaccine_id ASC
```

The 'Data Output' tab shows the results of the query:

vaccine_id	vaccine_price	doses
1	1	1500
2	2	700
3	3	200
4	4	350
5	5	6
6	6	2800
7	7	140
8	8	700
9	9	500

A green success message at the bottom right of the results area states: "Successfully run. Total query runtime: 207 msec. 9 rows affected."

SQL QURIES: -

1.) Number of customers taking vaccines whose age is from 19 to 40 based on their gender.

QUERY:

```
select count (*) as number, gender
from customers1 natural join customers2
where age>=19 and age<=40
group by gender
```

```

1 select count (*) as number, gender
2   from customers1 natural join customers2
3  where age > 19 and age <= 40
4 group by gender
5

```

number	gender
2	T
21	M
9	F

Successfully run. Total query runtime: 156 msec.
3 rows affected.

- 2.) Give customers first name, district and state where they reside who have registered for vaccination at vaccine centre_id = 2.

QUERY:

```

with inter (customer_id, district, state) as (
  select c_id, district, state
    from customers3 natural join registration
   where slot_id in (select slot_id from slots where centre_id = '2'))
  select first_name as customer_name, district, state
    from inter natural join customers1
   where "inter".customer_id = "customers1".c_id

```

```

1 with inter(customer_id, district, state) as(
2   select c_id, district, state
3     from customers3 natural join registration
4    where slot_id in( select slot_id
5                      from slots
6                     where centre_id = '2' ))
7
8   select first_name as customer_name, district, state
9     from inter natural join customers1
10   where "inter".customer_id = "customers1".c_id
11

```

customer_name	district	state
Gobind	Mulugu	Telangana
Jagiti	Gulbarga	Karnataka
Kaveri	West Imphal	Manipur
Ajit	Gandhinagar	Gujarat

3.) Print number of peoples taking vaccine with respect to their age, such that the output is in descending order of count and count is greater than 2.

QUERY:

```
select age, count(*)  
from customers2  
group by age  
having count(*) > 2  
order by count(*) desc
```

age	count
53	5
20	5
26	4
44	4
49	3
60	3
56	3
30	3
41	3
43	3
34	3

4.) Who is/are the female customer to book a vaccine having minimum age?

QUERY:

```
select first_name, age  
from customers1 natural join customers2  
where gender = 'F' and age in (  
    select min(age) from customers2 natural join customers1 where gender = 'F')
```

```

1 select first_name, age
2 from customers1 natural join customers2
3 where gender = 'F' and age in (
4
5
6
7
8
)

```

first_name	age
Diya	19
Priyanka	19

5.) Give full name of all customers who have registered for corona vaccine.

QUERY:

```

select distinct first_name, middle_name, surname
from customers1 natural join registration natural join vaccination
where vaccine_name = 'Corona'

```

```

1 select distinct first_name,middle_name,surname
2 from customers1 natural join registration natural join vaccination
3 where vaccine_name = 'Corona'
4

```

first_name	middle_name	surname
Pratap	Pravin	Bachchan
Savitri	Manu	Patel
Gotam	Jayadev	Patil
Dip	Neeraj	Sharma
Ankur	Hari	Anand
Manas	Anand	Rao
Pallev	Narayanan	Das
Jai	Karthik	Gadhvi
Apoorva	Shankar	Chaudhary
Bhavana	Jayanta	Jain
Kumaran	Indra	Chaudhuri

6.) Give full information of customers, sort them by customer_id in descending order.

QUERY:

```

select c_id, (first_name, middle_name, surname) as fullname, gender, birthdate, age, district, state,
       country
from customers1 natural join customers2 natural join customers3 natural join customers4
order by c_id desc

```

```

1 select c.c_id, (first_name,middle_name,surname) as fullname,gender,birthdate,age,district,state,country
2 from customers1 natural join customers2 natural join customers3 natural join customers4
3 order by c.c_id desc

```

c_id	fullname	gender	birthdate	age	district	state	country
68	(Ram,Manjeet,Kulkarni)	M	05-05-1995	26	Zunheboto	Nagaland	India
69	(Anku,Hari,Anand)	M	23-09-1994	27	Medak	Telangana	India
70	(Harsh,Srinivas,Sharma)	M	23-01-1995	26	Warangal	Andhra Pradesh	India
71	(Dip,Neeraj,Sharma)	M	13-06-2001	20	East Garo Hills	Meghalaya	India
72	(Gotam,Jayadev,Patil)	M	05-02-1991	30	Dhatali	Tripura	India
73	(Meera,Satish,Rao)	F	28-04-1968	53	Pune	Maharashtra	India
74	(Manjusha,Subrahmanyam,Misra)	F	06-03-1987	34	Jigital	Telangana	India
75	(Sanjita,Anand,Ahmed)	M	09-10-1977	44	Jhansi	Uttar Pradesh	India
76	(Manpreet,Sandeep,D'Cruze)	M	24-09-1968	53	Tirunelveli	Tamil Nadu	India
77	(Pratap,Pravin,Bachchan)	M	16-06-1997	24	Nalanda	Bihar	India
78	(Akash,Nirav,Vemulakonda)	M	28-02-1974	47	Jammu	Jammu and Kashmir	India
79	(Jay,Jitendra,Kulkarni)	M	04-02-1983	38	Changlang	Arunachal Pradesh	India
80	(Diya,Karthik,Bachchan)	F	04-02-2002	19	Jalandhar	Punjab	India

7.) Print customer name and id who has done online payment at centre with centre id = 10.

QUERY:

```

select first_name, c_id
from registration natural join slots natural join customers1
where payment_method = 'online' and centre_id = '10'

```

```

1 select first_name, c_id
2 from registration natural join slots natural join customers1
3 where payment_method = 'online' and centre_id = '10'

```

first_name	c_id
Harsh	204147348724
Gotam	184514202277

8.) How many manufacturers of vaccines are there in state of Maharashtra and Gujarat in total?

Query: -

```

select count (*) from manufacturer
where state='Gujarat' or state='Maharashtra'

```

The screenshot shows the pgAdmin 4 interface. The left sidebar displays a tree view of database objects under '201901224_db/postgres@PostgreSQL 13'. The 'Tables' node is expanded, showing various tables like centres1, centres2, centres3, customer, customer, customer, customer, employee, manufacturer, material, product, raw_mat, registration, roles, slots, supplies, vaccination, and vaccine. The 'Query Editor' tab is active, containing the SQL query:

```
1 select count(*) from manufacturer where state='Gujarat' or state='Maharashtra'
```

The results pane shows a single row of data:

	count
1	7

9.) How many customers have made the payment by using online platform?

Query: -

```
select count (*) from registration  
where payment_method='online'
```

The screenshot shows the pgAdmin 4 interface. The left sidebar displays a tree view of database objects under '201901224_db/postgres@PostgreSQL 13'. The 'Tables' node is expanded, showing various tables including registration, which is highlighted. The 'Query Editor' tab is active, containing the SQL query:

```
1 select count(*) from registration where payment_method='online'
```

The results pane shows a single row of data:

	count
1	50

10.) Which supplier supply more than 2 raw_materials?

Query :-

```
select s_id, s_name from raw_material  
where s_id in (select s_id from material_details  
group by s_id having count (*)>1)
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Toolbar:** File, Object, Tools, Help.
- Browser:** Shows database objects: centres2, centres3, customers1, customers2, customers3, customers4, employee_details, manufacturer, manufacturer_vaccine, material_details (selected), production, raw_material.
- Query Editor:** Contains the SQL query:

```
1 select s_id,s_name from raw_material where s_id  
2 in(select s_id from material_details group by s_id having count(*)>1)
```
- Data Output:** A table showing the results of the query:

s_id	s_name
1	Shiv Sales Corporation
2	AshChemie India
3	Triveni Chemicals
4	Ganesh Benzoplast Limited
5	Westman Chemicals Pvt. Ltd.
6	Healthy Life Pharma Private Limited
7	Gayati Poly Chem Private Limited
8	Khushi Chemical Pvt Ltd
9	Ultramarines India (P) Ltd
10	Penta Bioscience Products
11	Vats International
12	Prasol Chemicals Limited
13	Triveni Aromatics And Perfumery Private Limited
14	Ganesh Benzoplast Limited
- System Bar:** Shows the date and time (11/16/2021, 3:48 PM).

11.) Which state has the maximum no. of vaccines manufacturer?

Query: -

```
with inter (id, cou) as (select manufacturer_id, count (*)  
from manufacturer_vaccine  
group by manufacturer_id)  
select state from manufacturer where manufacturer_id  
in (select id from inter where cou= (select max(cou) from inter))
```

```

with inter(id,cou)as(select manufacturer_id,count(*) from manufacturer_vaccine
group by manufacturer_id)
select state from manufacturer where manufacturer_id
in(select id from inter where cou=(select max(cou) from inter))

```

12.) Which manufacturer produce maximum type of vaccine?

Query: -

```

with inter (id, cou) as (select manufacturer_id, count (*)
from manufacturer_vaccine group by manufacturer_id)
select manufacturer_id, company_name from manufacturer
where manufacturer_id in (select id from inter
where cou= (select max(cou) from inter))

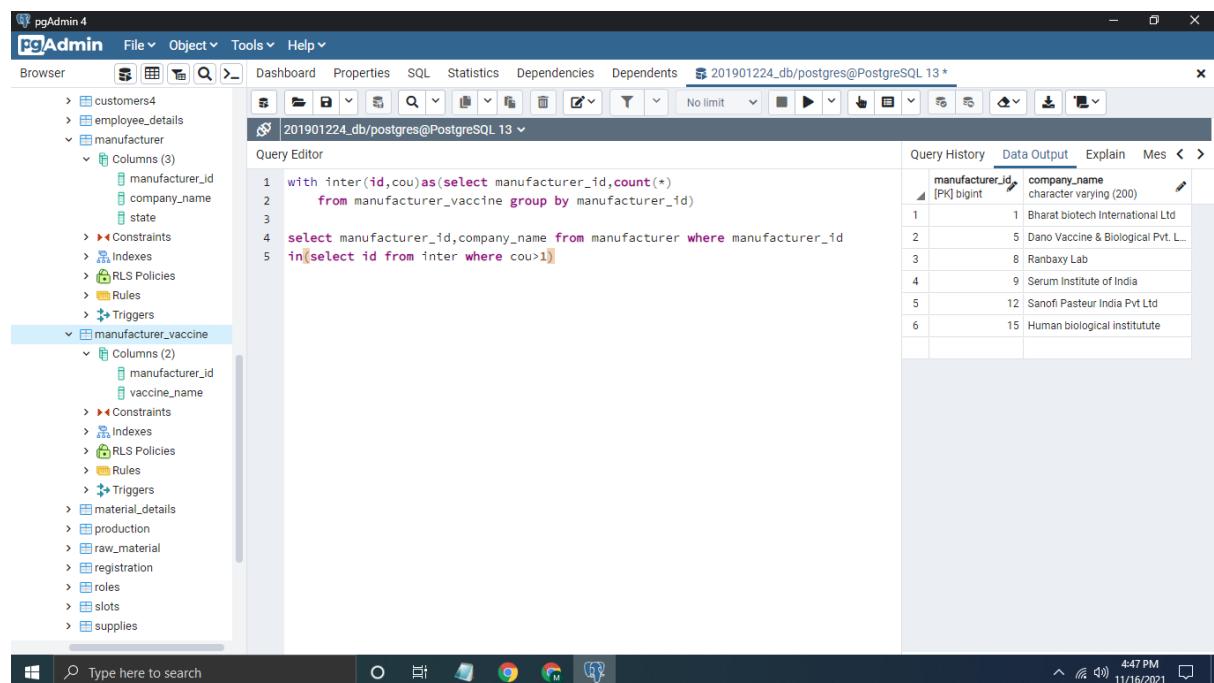
```

manufacturer_id	company_name
1	Bharat Biotech International Ltd

13.) which company produces more than 1 dose?

Query: -

```
with inter (id, cou) as (select manufacturer_id, count (*)
from manufacturer_vaccine group by manufacturer_id)
select manufacturer_id, company_name
from manufacturer where manufacturer_id
(Select id from inter where cou>1)
```



14.) Display Vaccine name and price in order of their price

Query: -

```
select vaccine_name, vaccine_price  
from vaccine1 natural join vaccine2  
order by vaccine_price desc
```

pgAdmin 4

201901224_db/postgres@PostgreSQL 13*

```

1 select vaccine_name,vaccine_price from
2 vaccine1 natural join vaccine2 order by vaccine_price desc

```

vaccine_name	vaccine_price
HPV	2800
Typhoid	1500
Rotavirus	900
BCG	700
Corona	700
Rabies	350
H1N1	200
Hep B Bulk	140
TT	6

15,) Which manufacturer produce maximum vaccines?

Query: -

with inter (id, sm) as

(Select manufacturer_id, sum (items)

from production group by manufacturer_id)

select manufacturer_id, company_name

from manufacturer where manufacturer_id

in (select manufacturer_id from manufacturer where manufacturer_id

in (select id from inter where sm= (select max(sm) from inter)))

pgAdmin 4

201901224_db/postgres@PostgreSQL 13*

```

1 with inter(id,sm)as
2 (select manufacturer_id,sum(items) from production group by manufacturer_id)
3
4 select manufacturer_id,company_name from manufacturer where manufacturer_id
5 in(select manufacturer_id from manufacturer where manufacturer_id
6 in(select id from inter where sm=(select max(sm) from inter)))
7

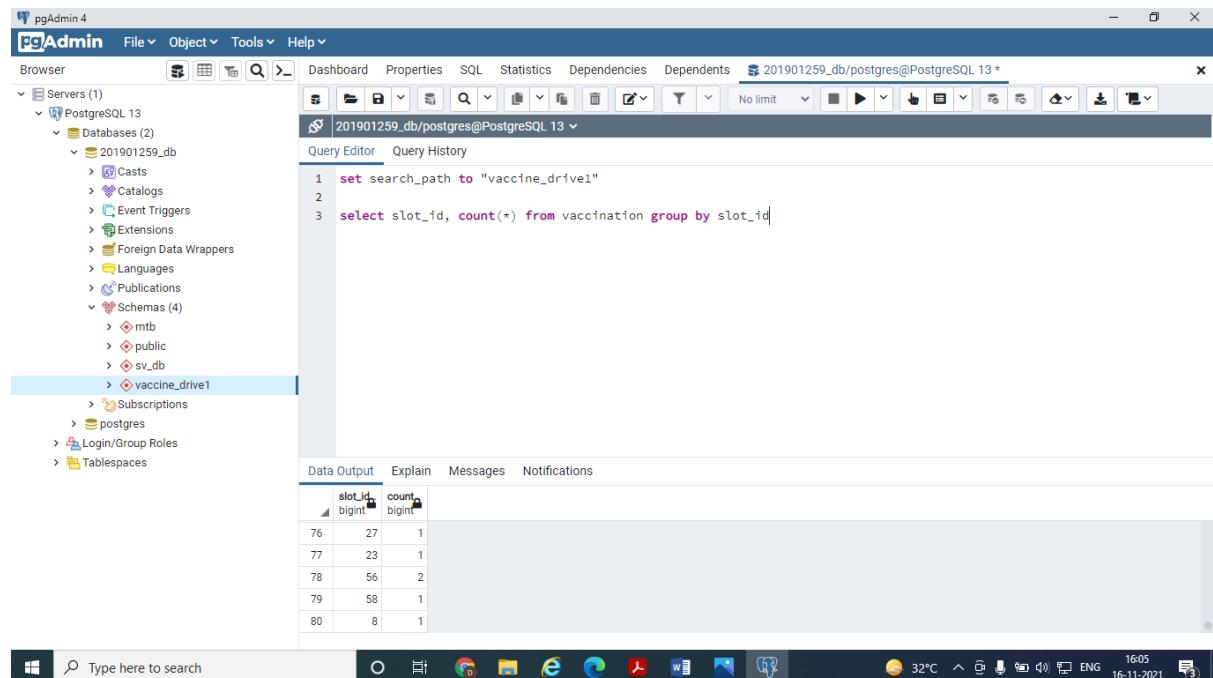
```

manufacturer_id	company_name
1	Bharat biotech International Ltd

16.) Find the number of vaccines available per slot

Query –

```
select slot_id, count(*)  
from vaccination  
group by slot_id
```



The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure under 'Servers'. A connection to '201901259_db' is selected. In the main area, the 'Query Editor' tab is active, showing the SQL query:

```
1 set search_path to "vaccine_drive1"  
2  
3 select slot_id, count(*) from vaccination group by slot_id
```

The results are displayed in a table titled 'Data Output'.

slot_id	count
76	27
77	23
78	56
79	58
80	8

17.) List the number of centres with respect to its category.

Query –

```
select category, count(*)  
from centres2  
group by category
```

pgAdmin 4

File ▾ Object ▾ Tools ▾ Help ▾

Browser Dashboard Properties SQL Statistics Dependencies Dependents 201901259_db/postgres@PostgreSQL 13 *

Query Editor Query History

```
1 set search_path to "vaccine_drive1"
2
3 Select category, count(*) from centres2 group by category;
```

Data Output Explain Messages Notifications

category	count
Private	7
Government	13

Type here to search

18.) List the centre_id, centre_name and category start with 'W'

Query -

```
select centre_id, centre_name, category
from centres2
where centre_name like 'W%'
```

pgAdmin 4

File ▾ Object ▾ Tools ▾ Help ▾

Browser Dashboard Properties SQL Statistics Dependencies Dependents 201901259_db/postgres@PostgreSQL 13 *

Query Editor Query History

```
1 set search_path to "vaccine_drive1"
2
3 select centre_id,centre_name,category from centres2 where centre_name like 'W%'
```

Data Output Explain Messages Notifications

centre_id	centre_name	category
3	West valley hospital	Private
4	Willow Green Hospital	Government
17	Winter warm	Government

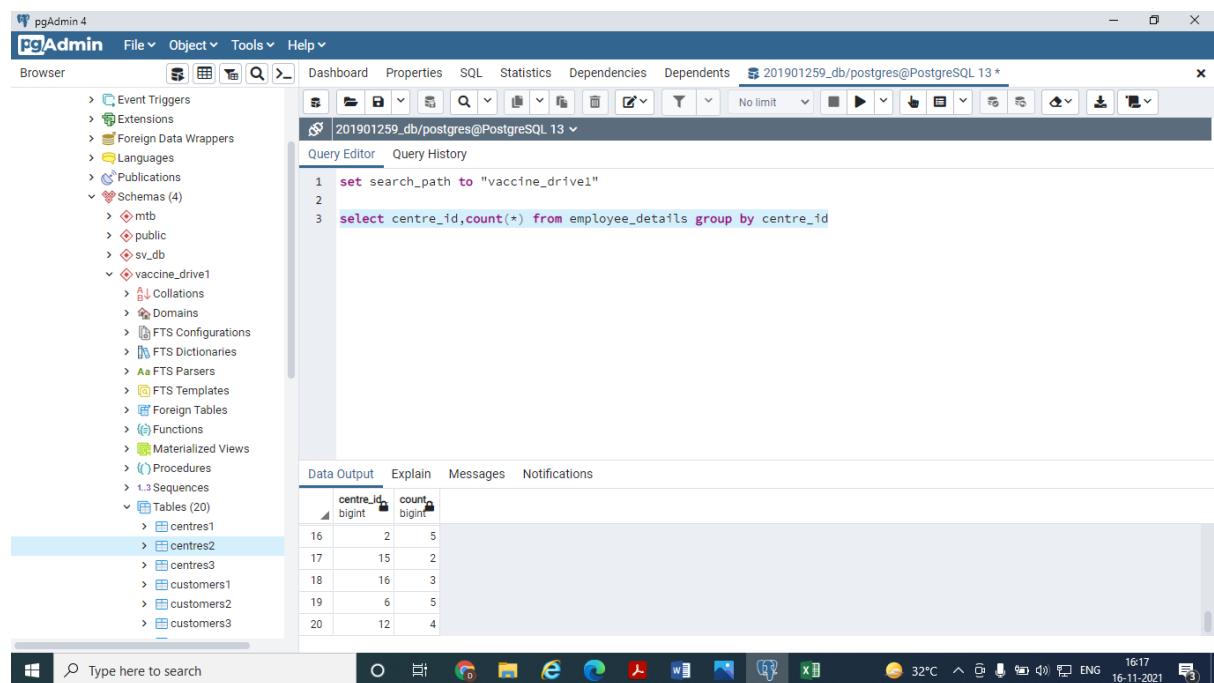
Successfully run. Total query runtime: 1 secs 352 msec. 3 rows affected.

Type here to search

19.) List the number of employees working in each centre.

Query -

```
select centre_id, count (*)
from employee_details
group by centre_id
```



The screenshot shows the pgAdmin 4 interface. On the left is the object browser tree, which includes Event Triggers, Extensions, Foreign Data Wrappers, Languages, Publications, Schemas (4), Schemas (4) expanded to show mtb, public, sv_db, and vaccine_drive1, which further expands to show Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Procedures, Sequences, Tables (20), and Tables (20) expanded to show centres1, centres2, centres3, customers1, customers2, and customers3. The central pane is the Query Editor with the following SQL code:

```
1 set search_path to "vaccine_drive1"
2
3 select centre_id, count(*) from employee_details group by centre_id
```

The Data Output tab shows the results of the query:

centre_id	count
16	2
17	15
18	16
19	6
20	12

20.) List the employee's name, id and their working location.

Query -

```
select e_id, first_name, district, state, country
from employee_details natural join centres1 natural join centres3
```

```

1 set search_path to "vaccine_drive1"
2
3 select e_id, first_name, district, state, country
4 from employee_details natural join
5 centres1 natural join centres3

```

e_id	first_name	district	state	country
68	68	Devika	Krishna	Andhra Pradesh
69	69	Kaur	Patna	Bihar
70	70	Purushottam	Mumbai City	Maharashtra
71	71	Jayant	North Goa	Goa
72	72	Dhananjay	Anand	Gujarat
73	73	Dipti	kra daadi	Arunachal Pradesh
74	74	Kalyana	Kurnool	Andhra Pradesh
75	75	Chand	Banaskantha	Gujarat
76	76	Jayant	Kurnool	Andhra Pradesh
77	77	Dhananjay	Patna	Bihar
78	78	Dipti	Banaskantha	Gujarat
79	79	Kalyana	Banaskantha	Gujarat
80	80	Chand	Mumbai City	Maharashtra

Explain Messages Notifications
Successfully run. Total query runtime: 312 msec.
80 rows affected.

21.) List the employee id and name and corresponding number of roles performing by them

Query -

```

with try (e_id, num_roles) as (
  select e_id, count (*)
    from roles
   group by e_id)

select e_id, first_name, num_roles
  from employee_details natural join try

```

```

1 set search_path to "vaccine_drive1"
2
3
4 with try(e_id,num_roles) as (
5 select e_id, count(*)
6 from roles
7 group by e_id )
8 select e_id, first_name, num_roles from employee_details natural join try

```

e_id	first_name	num_roles
65	Nanda	1
66	Deo	2
67	Rohan	1
68	Devika	1
69	Kaur	2
70	Purushottam	1
71	Jayant	1
72	Dhananjay	1
73	Dipti	2
74	Kalyana	1
75	Chand	1
76	Jayant	2
77	Dhananjay	1
78	Dipti	1
79	Kalyana	2

22.) Find the number of supply of vaccine done by each manufacturer to each centre in descending order and also print centre name.

Query -

```

with try (centre_id, supply) as (
    select centre_id, sum(supplied_count)
        from manufacturer natural join supplies
        group by centre_id)

```

```

select centre_id, centre_name, supply
    from centres2 natural join try
    order by supply desc

```

```

1 set search_path to "vaccine_drive1"
2
3 with try(centre_id,supply) as (
4 select centre_id, sum(supplied_count)
5 from manufacturer natural join supplies
6 group by centre_id )
7 select centre_id, centre_name, supply
8 from centres2 natural join try
9 order by supply desc
10

```

centre_id	centre_name	supply
3	15 Flowerflake Clinic	2041
4	10 Ablecare	1840
5	9 Injury Care Medics	1593
6	5 Hopewell Hospital	1388
7	8 Family Wellness Center	944
8	1 Delight Sun Clinic	841
9	7 Medica Zone	719
10	4 Willow Green Hospital	686
11	12 Ultra Care House	641
12	6 Community Health Service	528
13	2 Another life Hospital	512
14	14 Redstar Hospital	451
15	11 Blue Line Health Care Co.	329

Explain Messages Notifications
Successfully run. Total query runtime: 451 msec.
15 rows affected.

23.) Find the number of different manufacturer supply the vaccine with the specific centre and centre name.

Query -

```

with try (centre_id, count) as (
    select centre_id, count (*)
        from manufacturer natural join supplies
            group by centre_id)

select centre_name, count
    from try natural join centres2

```

```

1 set search_path to "vaccine_drive1"
2
3 with try(centre_id, count) as (
4 select centre_id, count(*)
5 from manufacturer natural join supplies
6 group by centre_id )
7 select centre_name, count
8 from try natural join centres2
9

```

centre_name	count
Delight Sun Clinic	1
Another life Hospital	1
West valley hospital	4
Willow Green Hospital	2
Hopewell Hospital	2
Community Health Service	2
Medica Zone	2
Family Wellness Center	2
Injury Care Medics	3
Ablecare	2
Blue Line Health Care Co.	2
Ultra Care House	1
Ace hospital	3
Redstar Hospital	1

Explain Messages Notifications
Successfully run. Total query runtime: 126 msec.
15 rows affected.

✓ Successfully run. Total query runtime: 126 msec. 15 rows affected.

24.) Find the number of different vaccine types available per slot in each centre.

Query -

```

with try (slot_id, count) as (
    select slot_id, count (*)
    from vaccination
    group by slot_id)

select centre_id, centre_name, slot_id, count
    from centres2 natural join slots natural join try

```

pgAdmin 4

File Object Tools Help

Browser

- > FTS Templates
- > Foreign Tables
- > Functions
- > Materialized Views
- > Procedures
- > Sequences
- > Tables (20)
 - > centres1
 - > centres2
 - > centres3
 - > customers1
 - > customers2
 - > customers3
 - > customers4
 - > employee_details
 - > Columns (5)
 - > Constraints
 - > Indexes
 - > RLS Policies
 - > Rules
 - > Triggers
 - > manufacturer
 - > manufacturer_vaccine
 - > material_details
 - > production
 - > raw_material
 - > registration
 - > Columns (3)

Query Editor Query History

```

1 set search_path to "vaccine_drive1"
2
3 with try(slot_id,count) as(
4 select slot_id, count(*)
5 from vaccination
6 group by slot_id )
7 select centre_id, centre_name, slot_id, count
8 from centres2 natural join slots natural join try

```

Data Output

centre_id	centre_name	slot_id	count
67	17 Winter warm	67	1
68	17 Winter warm	68	1
69	18 Angelwalk Hospital	69	1
70	18 Angelwalk Hospital	70	1
71	18 Angelwalk Hospital	71	1
72	18 Angelwalk Hospital	72	1
73	19 Intense Hospital	73	1
74	19 Intense Hospital	74	2
75	19 Intense Hospital	75	2
76	19 Intense Hospital	76	1
77	20 Summer spring Hospital	77	1
78	20 Summer spring Hospital	78	1
79	20 Summer spring Hospital	79	1
80	20 Summer spring Hospital	80	1

Explain Messages Notifications

Successfully run. Total query runtime: 183 msec.
80 rows affected.

Type here to search O Google Chrome Microsoft Edge File Explorer Task View XLSX 32°C ENG 16:56 16-11-2021

25.) Find the centre details of the centres which are in state Gujarat.

Query: -

```

select centre_id,centre_name,category, district
from centres1 natural join centres2
where state='Gujarat'

```

pgAdmin 4

File Object Tools Help

Browser

- > SV_LB
- > project
- > public
- > sv_pr
- > sv_pr
- > vaccine_drive
- > vaccine_drive1
 - > Collations
 - > Domains
 - > FTS Configurations
 - > FTS Dictionaries
 - > FTS Parsers
 - > FTS Templates
 - > Foreign Tables
 - > Functions
 - > Materialized Views
 - > Procedures
 - > Sequences
 - > Tables (20)
 - > centres1
 - > Columns (3)
 - centre_id
 - state
 - district
 - > Constraints
 - > Indexes
 - > RLS Policies
 - > Rules
 - > Triggers
 - > centres2
 - > Columns (3)
 - centre_id
 - centre_name
 - category

Query Editor Query History

```

1 select centre_id,centre_name,category,district
2 from centres1 natural join centres2
3 where state='Gujarat'

```

Data Output Explain Messages Notifications

centre_id	centre_name	category	district
1	3 West valley hospital	Private	Ahmedabad
2	9 Injury Care Medics	Government	Anand
3	14 Redstar Hospital	Government	Bhavnagar
4	17 Winter warm	Government	Banaskantha

Type here to search O Google Chrome Microsoft Edge File Explorer Task View 25°C, Smoke ENG 16:05 18-11-2021

26.) Find the company_name which is importing sodium and potassium chloride from the manufacturer.

Query:

```
with inter (s_id, m_id) as
(Select s_id, manufacturer_id
from material_details natural join raw_material
where material='Potassium chloride' or material='Sodium chloride')
select distinct(company_name)
from inter inner join manufacturer
on manufacturer_id=m_id
```

company_name
GSK Asia Pvt. Ltd.
Haffkine, Acharya Donde Marg
Pasteur Institute of India
Serum Institute of India
Shantha Biotechnics Ltd

27.) Find the centres which are having the highest working staff and centres should be handled by the government.

Query: -

```
select centre_id, count (*)
from employee_details natural join centres2
where category='Government'
group by centre_id
order by count (*) desc limit 1
```

```

1 select centre_id, count(*)
2 from employee_details natural join centres2
3 where category='Government'
4 group by centre_id
5 order by count(*) desc limit 1

```

centre_id	count
8	7

28.) Name the state which has the highest number of vaccinated people.

Query:

```

select state, count (*)
from registration natural join centres1 natural join slots
group by state
order by count (*) desc limit 1

```

```

1 select state, count(*)
2 from registration natural join centres1 natural join slots
3 group by state
4 order by count(*) desc limit 1
5
6

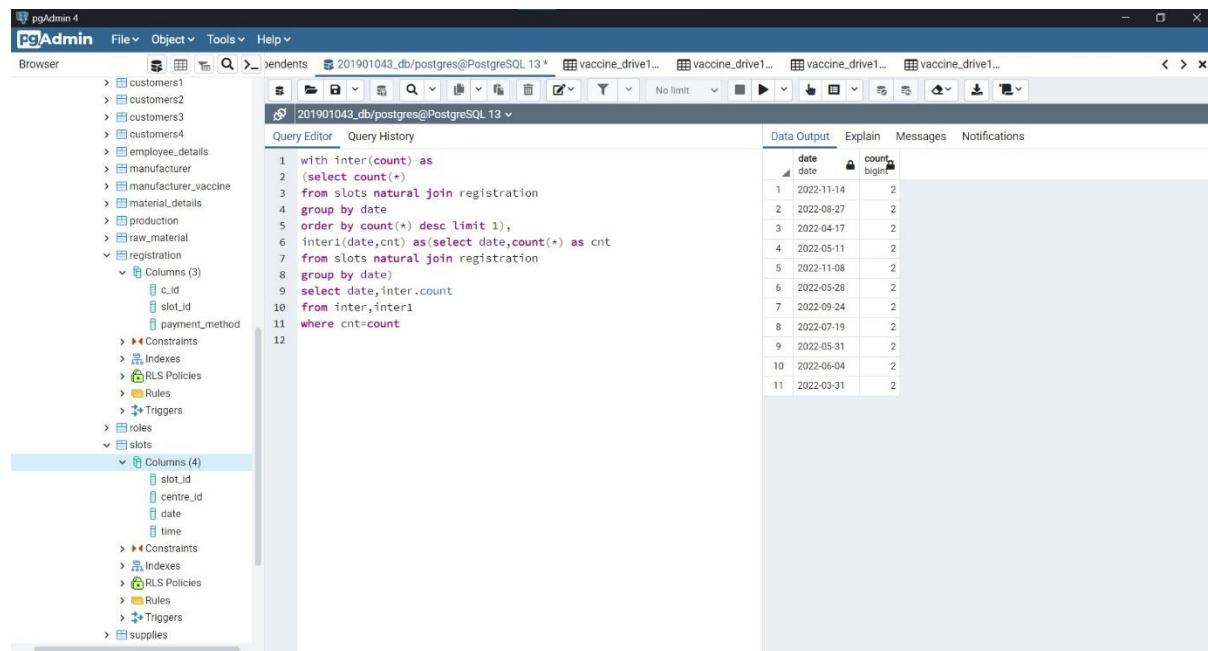
```

state	count
Arunachal Pradesh	23

29.) Find the date on which the highest number of registrations takes place.

Query:

```
with inter(count) as
  (Select count (*)
   from slots natural join registration
   group by date
   order by count (*) desc limit 1),
inter1(date, cnt) as (select date, count (*) as cnt
   from slots natural join registration
   group by date)
select date, inter.count
   from inter, inter1
  where cnt=count
```



The screenshot shows the pgAdmin 4 interface with a query editor window. The query is the one provided above, and the results are displayed in a table titled 'Data Output'.

date	count
2022-11-14	2
2022-08-27	2
2022-04-17	2
2022-05-11	2
2022-11-08	2
2022-05-28	2
2022-09-24	2
2022-07-19	2
2022-05-31	2
2022-06-04	2
2022-03-31	2

30.) Find the 4th,5th and 6th costliest vaccines that are sold at the vaccine centre.

Query: -

```

select vaccine_id, vaccine_name, vaccine_price
from vaccine2 natural join vaccine1
order by vaccine_price desc offset 3 rows fetch next 3 rows only

```

vaccine_id	vaccine_name	vaccine_price
1	8 BCG	700
2	2 Corona	700
3	4 Rabies	350

31.) List the number of vaccines taken by customers in descending order.

Query:

```

with inter (c_id, cnt) as
(Select c_id, count (*)
from registration
group by c_id)
select *
from customers1 natural join inter
order by cnt desc

```

pgAdmin 4

File Object Tools Help

Browser 201901043_db/postgres@PostgreSQL 13*

Query Editor Query History Data Output Explain Messages Notifications

```

1 with inter(c_id,cnt) as
2 (select c_id,count(*) as
3 from registration
4 group by c_id)
5 select *
6 from customers1 natural join inter
7 order by cnt desc
  
```

c_id	first_name	middle_name	surname	gender	cnt
1	Pallav	Narayanan	Das	M	3
2	Savitri	Manu	Patel	F	3
3	Gobind	Chandrakant	Sharma	M	3
4	Dip	Neeraj	Sharma	M	3
5	Chetana	Sushil	Singh	F	3
6	Jai	Karthik	Gadhwani	M	3
7	Jagjit	Dharma	Jain	M	2
8	Apoorva	Shankar	Chaudhary	F	2
9	Navin	Pravin	Singh	M	2
10	Sanjit	Anand	Ahmad	M	2
11	Ranjeet	Mohandas	Tamboli	M	2
12	Rakhi	Balwinder	Jain	F	2
13	Prashant	Devaraj	Nilhanupudi	M	2
14	Kumari	Rajiv	DCruze	F	2
15	Ankur	Hari	Anand	M	2
16	Viraj	Akash	Misra	M	2
17	Kaveri	Jagdish	DCruze	F	2
18	Suraj	Mahavir	Nagarkar	M	2
19	Bhavana	Jayanta	Jain	F	2
20	Jaidev	Brijesh	Kumar	M	2
21	Pratap	Pravin	Bachchan	M	2
22	Harsh	Srinivas	Sharma	M	2
23	Manjusha	Subrahmanyam	Misra	F	2
24	Prateek	Jitendra	Korrapati	M	2

32.) List the vaccine name and the number of customers taken that vaccine in ascending order.

Query:

```

with inter (vaccine_id, cnt) as
(Select vaccine_id, count (*) as cnt
from vaccination
group by vaccine_id)
select vaccine_name, vaccine_id, cnt
from vaccine1 natural join inter
order by cnt
  
```

pgAdmin 4

File Object Tools Help

Browser 201901043_db/postgres@PostgreSQL 13*

Query Editor Query History Data Output Explain Messages Notifications

```

1 with inter(vaccine_id,cnt) as
2 (select vaccine_id,count(*) as cnt
3 from vaccination
4 group by vaccine_id)
5 select vaccine_name,vaccine_id,cnt
6 from vaccine1 natural join inter
7 order by cnt
  
```

vaccine_name	vaccine_id	cnt
BCG	8	4
HPV	6	7
Rotavirus	9	8
TT	5	8
Typhoid	1	9
HINI	3	9
Corona	2	12
Rabies	4	13
Hep B Bulk	7	19

33.) List all vaccine_details and stock produced by the company with highest production.

QUERY:

```
select company_name, vaccine_name, vaccine_id, items  
from production  
where company_name in (select company_name  
from production  
group by company_name  
order by sum(items) desc limit 1)
```

The screenshot shows the pgAdmin 4 interface. The left sidebar (Browser) displays database objects: customers1, customers2, customers3, customers4, employee_details, manufacturer, manufacturer_vaccine, production, material_details, and vaccine_drive. The production object is selected. The right pane contains the Query Editor with the query provided above, and the Data Output tab showing the results of the query execution.

company_name	vaccine_name	vaccine_id	Items
Bharat biotech International Ltd	Rotavirus	9	3852
Bharat biotech International Ltd	Corona Vaccine	2	5986
Bharat biotech International Ltd	Typhoid	1	7844

34.) List the location (district, state) of 10 eldest person to get vaccinated with their age and customers id mentioned.

QUERY:

```
select c_id, age, district, state  
from customers2 natural join customers3  
order by age desc limit 10
```

```

1 select c_id, age, district, state
2 from customers2 natural join customers3
3 order by age desc limit 10

```

c_id	age	district	state
60238543934	70	Palakkad	Kerala
715236878273	67	Bhiwani	Haryana
997979339316	67	Gulbarga	Karnataka
317974741209	66	Medak	Telangana
595652905043	66	Jaunpur	Uttar Pradesh
543685173979	65	Champai	Mizoram
733164369775	63	Zunheboto	Nagaland
8538601193320	63	Jalandhar	Punjab
809187682257	61	Giridih	Jharkhand
551525925496	60	Srinagar	Jammu and Kashmir

Successfully run. Total query runtime: 79 msec. 10 rows affected.

35.) List all the customer's name who have taken vaccines which contains at least one raw material supplied by company located in 'Gujarat'.

QUERY:

```

with inter(manufacturer_id) as (select manufacturer_id
from raw_material where state = 'Gujarat')
select c_id, first_name
from customers1 natural join registration
where slot_id in (select slot_id from vaccination natural join inter)

```

```

1 with inter(manufacturer_id) as (select manufacturer_id
2 from raw_material
3 where state = 'Gujarat')
4
5 select c_id, first_name
6 from customers1 natural join registration
7 where slot_id in (select slot_id
8 from vaccination natural join inter)
9

```

c_id	first_name
538601193320	Navneet
506239648822	Akbar
679543418497	Jai
317974741209	Parminder
132885012231	Akash
777691027727	Kumari
628088809326	Vikram
513652206329	Mitra

36.) Find the fifth to ten highest number of customers visited centre id, centre name and centre location.

Query:

```
with try (slot_id, count) as (
    select slot_id, count(*)
        from registration
       group by slot_id)

    select centre_id, centre_name, district, state, country, count
        from try natural join slots natural join centres1 natural join centres2 natural join centres3
    order by count desc offset 4 rows fetch next 5 rows only
```

The screenshot shows the PgAdmin 4 interface with the following details:

- Browser:** Shows the schema structure under "vaccine_drive1".
- Query Editor:** Contains the SQL query provided above.
- Messages:** Displays "Successfully run. Total query runtime: 154 msec." and "5 rows affected."
- Data Output:** A table showing the results of the query. The columns are: centre_id, centre_name, district, state, country, and count.

	centre_id	centre_name	district	state	country	count
1	6	Community Health Service	South Goa	Goa	India	2
2	18	Angenwati Hospital	Trap	Arunachal Pradesh	India	2
3	9	Injury Care Medics	Anand	Gujarat	India	1
4	17	Winter warm	Banaskantha	Gujarat	India	1
5	8	Family Wellness Center	Kurnool	Andhra Pradesh	India	1

37.) Find the revenue of all the centres in descending order and list centre id, name and revenue?

Query -

```
with try (centre_id, sum) as (
    select centre_id, sum(vaccine_price)
        from slots natural join vaccination natural join vaccine2
       group by centre_id
    )

    select centre_id, centre_name, sum
        from centres2 natural join try
    order by sum desc
```

```

set search_path to "vaccine_drive1"
with try(centre_id,sum) as(
select centre_id,sum(vaccine_price)
from slots natural join vaccination natural join vaccine
group by centre_id
)
select centre_id,centre_name,sum
from centres2 natural join try
order by sum desc;

```

centre_id	centre_name	sum
1	Willow Green Hospital	6990
2	Hopewell Hospital	6250
3	Redstar Hospital	4540
4	Injury Care Medics	4506
5	Ablecare	4206
6	Summer spring Hospital	4050
7	Community Health Service	3900
8	Intense Hospital	3530
9	Ultra Care House	3436
10	Angelwalk Hospital	3146
11	Winter warm	2650
12	Ace hospital	2146
13	Family Wellness Center	2090
14	Delight Sun Clinic	1940
15	West valley hospital	1756
16	lifeflash Clinic	1320
17	Flowerlake Clinic	1256
18	Blue Line Health Care Co.	1180
19	Medica Zone	1180
20	Another life Hospital	486

38) Which manufacturer produce maximum vaccines?

Query: -

with inter (id, sm) as

(Select manufacturer_id, sum(items) from production group by manufacturer_id)
 select manufacturer_id, company_name from manufacturer where manufacturer_id
 in (select manufacturer_id from manufacturer where manufacturer_id
 in (select id from inter where sm= (select max(sm) from inter)))

```

with inter(id,sm) as
(select manufacturer_id,sum(items) from production group by manufacturer_id)
select manufacturer_id,company_name from manufacturer where manufacturer_id
in(select manufacturer_id from manufacturer where manufacturer_id
in(select id from inter where sm=(select max(sm) from inter)))

```

manufacturer_id	company_name
1	Bharat biotech International Ltd

39.) Customers booking slots in morning before 12 am

Query -

select c_id, first_name, middle_name, surname, slot_id, time

from slots natural join registration natural join customers1
where time like '%10%'

c_id	first_name	middle_name	surname	slot_id	time
29	Jaidev	Brijesh	Kumar	70	10:00 - 12:00
30	Shresth	Rajneesh	Ahmad	29	8:00 - 10:00
31	Kaveri	Jagadish	D'Cruze	6	10:00 - 12:00
32	Ranjeet	Mohandas	Tamboli	54	10:00 - 12:00
33	Narinder	Nand	Joshi	49	8:00 - 10:00
34	Ajit	Vivek	Misra	5	8:00 - 10:00
35	Abhay	Om	Ahmed	14	10:00 - 12:00
36	Prashant	Devaraj	Nibhanupudi	61	8:00 - 10:00
37	Bhavna	Jayanta	Jain	70	10:00 - 12:00
38	Dip	Neeraj	Sharma	41	8:00 - 10:00
39	Chander	Sarvesh	Ahmed	57	8:00 - 10:00
40	Prateek	Jitendra	Korrapati	30	10:00 - 12:00
41	Navneet	Harshal	Kulkarni	2	10:00 - 12:00
42	Kumari	Rajiv	D'Cruze	18	10:00 - 12:00
43	Mitra	Harendra	Vemulakonda	78	10:00 - 12:00

40.) Find the number of vaccines at centre_id = 5 in different slots and sort it with respect to the number of vaccines available in that slot.

Query -

```
with try (slot_id, count) as (
  select slot_id, count(*)
  from vaccination
  group by slot_id
)
select centre_id, slot_id, count
from slots natural join try
where centre_id = 5
```

Trigger Function: -

Create a trigger which assign the role of doctor to the employee which was inserted into employee_details table and add it to the roles table. Trigger should automatically update the role records when any insert take place on employee_details table.

Query: -

```

begin
insert into "vaccine_drive1".Roles (e_id, role) values(new.e_id,'Doctor');
return New;
end
    
```

pgAdmin 4

File Object Tools Help

Browser

- > centres2
- > centres3
- > customers1
- > customers2
- > customers3
- > customers4
- > employee_details
 - > Columns (5)
 - > Constraints
 - > Indexes
 - > RLS Policies
 - > Rules
 - > Triggers (1)
- > manufacturer
- > manufacturer_vaccine
- > material_details
- > production
- > raw_material
- > registration
- > roles
- > slots
- > supplies
- > vaccination
- > vaccine1
- > vaccine2
- > Trigger Functions (1)
 - (e_employee_role)
- > Types
- > Views
- > Subscriptions
- > postgres
- > Login/Group Roles
- > Tablespaces

201901043_db/postgres@PostgreSQL 13

Query Editor Query History

```

1 INSERT INTO vaccine_drive1.employee_details(
2   e_id, first_name, middle_name, surname, centre_id)
3     VALUES (81, 'Roshani', 'Rajendra', 'Patel', 3);
  
```

Data Output Explain Messages Notifications

INSERT 0 1

Query returned successfully in 251 msec.

pgAdmin 4

File Object Tools Help

Browser

- > centres2
- > centres3
- > customers1
- > customers2
- > customers3
- > customers4
- > employee_details
 - > Columns (5)
 - > Constraints
 - > Indexes
 - > RLS Policies
 - > Rules
 - > Triggers (1)
- > manufacturer
- > manufacturer_vaccine
- > material_details
- > production
- > raw_material
- > registration
- > roles
- > slots
- > supplies
- > vaccination
- > vaccine1
- > vaccine2
- > Trigger Functions (1)
 - (e_employee_role)
- > Types
- > Views
- > Subscriptions
- > postgres
- > Login/Group Roles
- > Tablespaces

201901043_db/postgres@PostgreSQL 13

Query Editor Query History

```

1 SELECT * FROM vaccine_drive1.roles
2 ORDER BY e_id ASC, role ASC
  
```

Data Output Explain Messages Notifications

e_id	role
70	Helper
71	Helper
72	Nurse
73	Nurse
74	Helper
75	Helper
76	Nurse
77	Manager
78	Nurse
79	Doctor
80	Helper
81	Nurse
82	Nurse
83	Helper
84	Doctor
85	Manager
86	Helper
87	Helper
88	Doctor
89	Manager
90	Helper
91	Doctor

Stored Procedure: -

Create a simple stored procedure to list all records as a message of customers1 table.

Query: -

DECLARE

R_LIST2 record;

BEGIN

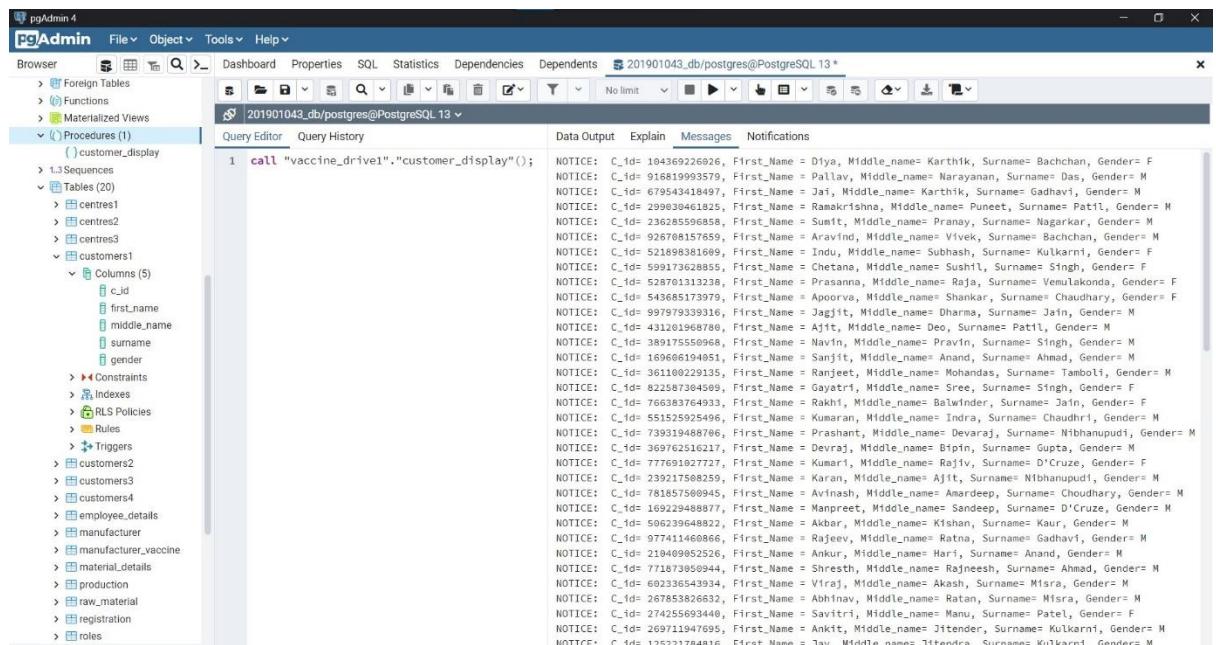
FOR R_LIST2 in (select c_id, first_name, middle_name, surname, gender
from vaccine_drive1.customers1)

loop

RAISE NOTICE 'C_id= %, First_Name = %, Middle_name= %,
Surname= %, Gender= %', R_LIST2.c_id, R_LIST2.first_name,
R_LIST2.middle_name, R_LIST2.surname, R_LIST2.gender;

end loop;

END;



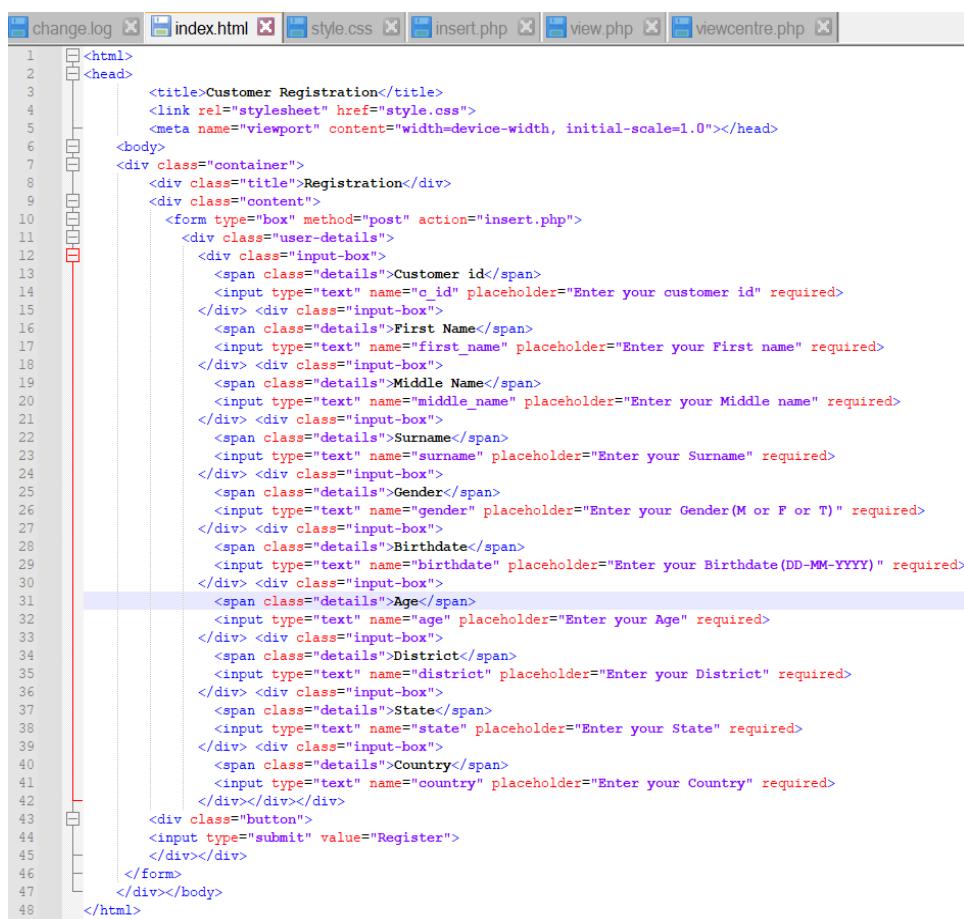
```
1  call "vaccine_drive1"."customer_display"();
```

C_id	First_Name	Middle_name	Surname	Gender
104369226026	Diya	Karthik	Bachchan	F
918819993579	Pallav	Narayanan	Das	M
679543418497	Jai	Karthik	Gadhavi	M
299639461825		Ramakrishna	Puneet	M
236285596858	Sumit	Pranay	Nagarkar	M
926768157659	Aravind	Vivek	Bachchan	M
521698381689	Indu	Subhash	Kulkarni	F
599173628855	Chetana	Sushil	Singh	F
528701313238	Prasanna	Raja	Venulakonda	F
543685173979	Apoorva	Shankar	Chaudhary	M
997979339316	Dharma	Jatin		M
431201968789	Ajita	Deo	Patil	M
389175550968	Naresh	Pravin	Singh	M
169606194051	Sanjiv	Anand	Ahmed	M
361108229135	Ranjeet	Mohandas	Tamboli	M
822587304509	Gayatri	Sree	Singh	F
766383764933	Rakhi	Balwinder	Jain	F
551525925496	Kumaran	Indra	Chaudhri	M
739319488706	Prashant	Devaraj	Nithanupudi	M
369762516217	Devraj	Bipin	Gupta	M
777691827727	Kumar	Rejiv	D'Cruze	F
239217568259	Karan	Ajith	Nithanupudi	M
781857580945	Avinash	Aorderdeep	Choudhary	M
16922948877	Manpreet	Sandeep	D'Cruze	M
596239648822	Akbar	Kishan	Kaur	M
977411460866	Rajeev	Ratna	Gadhavi	M
219409052526	Ankur	Hari	Anand	M
771873050944	Shresth	Rajneesh	Ahmed	M
602336543934	Vital	Aakash	Misra	M
267853826632	Abhinav	Ratan	Misra	M
274255693440	Savitri	Manu	Patel	F
269711947695	Ankit	Jitender	Kulkarni	M
125221784816	Jay	Jitendra	Kulkarni	M

SECTION7: PROJECT CODE WITH OUTPUT SCREENSHOTS

- We made a webpage from the database with the help of Xampp software and Writing codes in HTML, CSS, and PHP languages.
- At first, the website is designed by index.html and styles.css which then after uses other files which are insert.php to register customers.

❖ Code of Index.html file:



```

1 <html>
2   <head>
3     <title>Customer Registration</title>
4     <link rel="stylesheet" href="style.css">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0"></head>
6   <body>
7     <div class="container">
8       <div class="title">Registration</div>
9       <div class="content">
10      <form type="box" method="post" action="insert.php">
11        <div class="user-details">
12          <div class="input-box">
13            <span class="details">Customer id</span>
14            <input type="text" name="c_id" placeholder="Enter your customer id" required>
15          </div> <div class="input-box">
16            <span class="details">First Name</span>
17            <input type="text" name="first_name" placeholder="Enter your First name" required>
18          </div> <div class="input-box">
19            <span class="details">Middle Name</span>
20            <input type="text" name="middle_name" placeholder="Enter your Middle name" required>
21          </div> <div class="input-box">
22            <span class="details">Surname</span>
23            <input type="text" name="surname" placeholder="Enter your Surname" required>
24          </div> <div class="input-box">
25            <span class="details">Gender</span>
26            <input type="text" name="gender" placeholder="Enter your Gender(M or F or T)" required>
27          </div> <div class="input-box">
28            <span class="details">Birthdate</span>
29            <input type="text" name="birthdate" placeholder="Enter your Birthdate(DD-MM-YYYY)" required>
30          </div> <div class="input-box">
31            <span class="details">Age</span>
32            <input type="text" name="age" placeholder="Enter your Age" required>
33          </div> <div class="input-box">
34            <span class="details">District</span>
35            <input type="text" name="district" placeholder="Enter your District" required>
36          </div> <div class="input-box">
37            <span class="details">State</span>
38            <input type="text" name="state" placeholder="Enter your State" required>
39          </div> <div class="input-box">
40            <span class="details">Country</span>
41            <input type="text" name="country" placeholder="Enter your Country" required>
42          </div></div></div>
43        <div class="button">
44          <input type="submit" value="Register">
45        </div></div>
46        </form>
47      </div></body>
48    </html>

```

- After the input is provided if the customer_id is unique and other conditions are correct than data is inserted in database with the help of insert.php file.

❖ Code of Insert.php file:

```

1 <?php
2 <html>
3   <body>
4     <button onclick="window.location.href = 'view.php';">
5       Customers details
6     </button>
7     <button onclick="window.location.href = 'viewcentre.php';">
8       Centre details
9     </button>
10    </body>
11  </html>
12 <?php
13
14 $host = "localhost";
15 $user = "postgres";
16 $pass = "soham";
17 $db = "201901274_db";
18
19 $con = pg_connect("host=$host dbname=$db user=$user password=$pass")
20         or die ("Could not connect to server\n");
21
22 if(!$con)
23 {
24   echo "Unable to connect with database\n";
25 }
26 else
27 {
28   $c_id = $_POST['c_id'];
29   $first_name = $_POST['first_name'];
30   $middle_name = $_POST['middle_name'];
31   $surname = $_POST['surname'];
32   $gender = $_POST['gender'];
33   $birthdate = $_POST['birthdate'];
34   $age = $_POST['age'];
35   $state = $_POST['state'];
36   $district = $_POST['district'];
37   $country = $_POST['country'];
38
39   $query = "INSERT into vaccine_drive_final.customers1(c_id,first_name,middle_name,surname,gender) VALUES('$c_id','$f
40   $query1 = "INSERT into vaccine_drive_final.customers2(c_id,birthdate,age) VALUES('$c_id','$birthdate','$age')";
41   $query2 = "INSERT into vaccine_drive_final.customers3(c_id,district,state) VALUES('$c_id','$district','$state')";
42   $query3 = "INSERT into vaccine_drive_final.customers4(c_id,state,country) VALUES('$c_id','$state','$country')";
43   $result = pg_query($con,$query);
44   $result = pg_query($con,$query1);
45   $result = pg_query($con,$query2);
46   $result = pg_query($con,$query3);
47   echo "Registration Done !!!!";
48
49 pg_close($con);
50 ?>
```

- After that the customer can view customer details and centres details with help of view.php and view centre.php files.

❖ Code of view.php file:

```

1 <?php
2 <html>
3   <head>
4     <style>
5       table (
6         border-collapse: collapse;
7         width: 100%;
8       )
9       th, td {
10         padding: 8px;
11         text-align: left;
12         border-bottom: 1px solid #DDD;
13       }
14       tr:hover {background-color: #D9E9F7;}
15     </style>
16   <table border="1">
17     <tr>
18       <td>Customer id</td><td>First Name</td><td>Middle Name</td><td>Surname</td><td>
19       <td>Gender</td><td>Birthdate</td><td>Age</td><td>District</td><td>State</td><td>Country</td>
20     </tr>
21     <tr>
22       $host = "localhost";
23       $user = "postgres";
24       $pass = "soham";
25       $db = "201901274_db";
26       $con = pg_connect("host=$host dbname=$db user=$user password=$pass")
27           or die ("Could not connect to server\n");
28       $query="select c_id,first_name,middle_name,surname,gender,birthdate,age,district,state,country from vaccine_drive_final.custo
29       $result=pg_query($con,$query);
30       while($row=pg_fetch_array($result))
31       {
32         echo "<td>".$row['c_id']."</td>";
33         echo "<td>".$row['first_name']."</td>";
34         echo "<td>".$row['middle_name']."</td>";
35         echo "<td>".$row['surname']."</td>";
36         echo "<td>".$row['gender']."</td>";
37         echo "<td>".$row['birthdate']."</td>";
38         echo "<td>".$row['age']."</td>";
39         echo "<td>".$row['district']."</td>";
40         echo "<td>".$row['state']."</td>";
41         echo "<td>".$row['country']."</td>";
42       }
43     </tr>
44   pg_close($con);
45 ?>
46 </table>
47 </body>
48 </html>
```

❖ Code of viewcentre.php file:

```
1 <html>
2   <body>
3     <style>
4       table {
5         border-collapse: collapse;
6         width: 100%;
7       }
8       th, td {
9         padding: 8px;
10        text-align: left;
11        border-bottom: 1px solid #DDD;
12      }
13      tr:hover {background-color: #D6EEEE;}
14    </style>
15    <table border="2">
16      <tr>
17        <td><b>Centre id</b></td>
18        <td><b>Centre Name</b></td>
19        <td><b>Category</b></td>
20        <td><b>District</b></td>
21        <td><b>State</b></td>
22        <td><b>Country</b></td>
23      </tr>
24      <?php
25        $host = "localhost";
26        $user = "postgres";
27        $pass = "soham";
28        $db = "201901274_db";
29        $con = pg_connect("host=$host dbname=$db user=$user password=$pass")
30        or die ("Could not connect to server\n");
31        $query1="select centre_id,centre_name,category,district,state,country from vaccine_drive_final.cer";
32        $result=pg_query($con,$query1);
33        while($row=pg_fetch_array($result))
34        {
35          echo "<td>".$row['centre_id']."</td>";
36          echo "<td>".$row['centre_name']."</td>";
37          echo "<td>".$row['category']."</td>";
38          echo "<td>".$row['district']."</td>";
39          echo "<td>".$row['state']."</td>";
40          echo "<td>".$row['country']."</td>";
41          echo "</tr>";
42        }
43        pg_close($con);
44      ?>
45    </table>
46  </body>
47 . . .
```

❖ Outputs

→ Initial webpage:

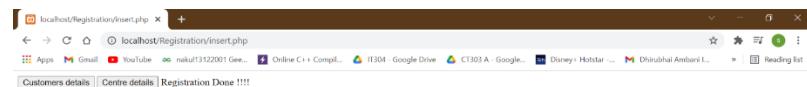
Customer Registration

Registration

Customer id	First Name
123456789012	Soham
Middle Name	Surname
Bhaveshkumar	Patel
Gender	Birthdate
M	12-03-2002
Age	District
19	Ahmedabad
State	Country
Gujarat	India

Register

→ After registration is completed successfully.



- If customer wants to see customers_details.
- Even we can see that all registered customers data is visible.

Customer ID	First Name	Last Name	Address	Gender	Date of Birth	Age	City	State	Country
692489279159	Prateek	Jitendra	Korrapati	M	18-06-1961	60	Belgaum	Karnataka	India
513652206329	Mitra	Harendra	Vemulakonda	F	15-07-1991	30	Tinsukia	Assam	India
960238912797	Sunil	Sankar	Sharma	M	11-04-1989	32	Satna	Madhya Pradesh	India
716354550365	Manas	Anand	Rao	M	11-07-2001	20	West District	Sikkim	India
411500130417	Priyanka	Lalit	Chandri	F	18-02-2002	19	Pakur	Jharkhand	India
278117849544	Abhay	Om	Ahmed	M	20-11-1971	49	Champai	Mizoram	India
715236876273	Shanta	Subrahmany	Jain	F	15-04-1954	67	Bhiwani	Haryana	India
601235392435	Rakesh	Shakti	Kumar	M	28-09-1966	55	Gwalior	Madhya Pradesh	India
477903374045	Ajit	Vivek	Misra	M	28-04-1968	53	Gandhinagar	Gujarat	India
628088809326	Vikram	Hardeep	Gadhavi	M	19-04-2001	20	Hisar	Haryana	India
184514202277	Gotam	Jayadev	Patil	M	05-02-1991	30	Dhalai	Tripura	India
455370152896	Sunil	Prashant	Chaudhary	M	22-03-1977	44	Jajpur	Rajasthan	India
605670716923	Dileep	Ram	Vemulakonda	M	18-09-1982	39	Ahmedabad	Gujarat	India
132885012231	Aakash	Nirav	Vemulakonda	M	28-02-1974	47	Jammu	Jammu and Kashmir	India
211824590716	Ram	Manjeet	Kulkarni	M	05-05-1995	26	Zunheboto	Nagaland	India
123456789012	Soham	Bhaveshkumar	Patel	M	12-03-2002	19	Ahmedabad	Gujarat	India

→ If customers want to see centre details.

Centre ID	Centre Name	Category	District	State	Country
1	Delight Sun Clinic	Government	Araria	Bihar	India
2	Another life Hospital	Government	North Goa	Goa	India
3	West valley hospital	private	Ahmedabad	Gujarat	India
4	Willow Green Hospital	Government	Krishna	Andhra Pradesh	India
5	Hopewell Hospital	Government	Ahmednagar	Maharashtra	India
6	Community Health Service	private	South Goa	Goa	India
7	Medica Zone	private	Lohit	Arunachal Pradesh	India
8	Family Wellness Center	Government	Kurnool	Andhra Pradesh	India
9	Injury Care Medics	Government	Anand	Gujarat	India
10	Ablecare	private	Visakhapatnam	Andhra Pradesh	India
11	Blue Line Health Care Co.	Government	kra daadi	Arunachal Pradesh	India
12	Ultra Care House	Government	Tawang	Arunachal Pradesh	India
13	Ace hospital	Government	Visakhapatnam	Andhra Pradesh	India
14	Redstar Hospital	Government	Bhavnagar	Gujarat	India
15	Flowerflake Clinic	Government	Shiyomi	Arunachal Pradesh	India