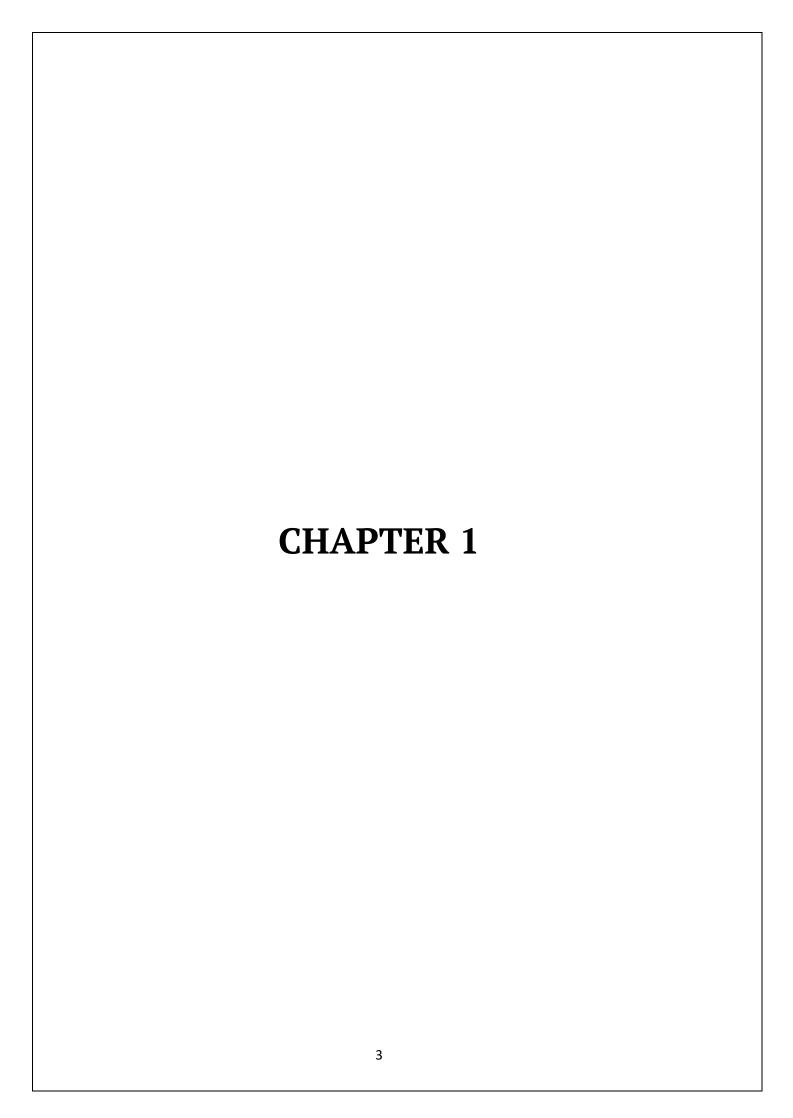# HUMAN RESOURCE MANAGEMENT AND PAYROLL SYSTEM

SUBMITTED BY :-

1.Vedika Patil (46)

2.Ramya Rajeshwari(50)

3.Pranjali Shelke(58)

INDEX

CHAPTER 1

INTRODUCTION:

Nowadays companies try to manage their enhancement and have better control over their employees. HRMS refers to the systems and processes at the intersection between human resource management(HRM) and information technology. The Human Resource Management System HRMS can be used to manage the Human Resource of a company.

The system helps a company to create vacancies and accept applications from the applicants, online. It also keeps the administrator updated with the progress of a team in the task assigned. It can also generate payroll automatically. The HRMS is a web based application and it can be accessed over the internet. Job seekers can view job vacancies and they can apply for the jobs. Employees, HR team and administrator can login to the system by entering Login Id and password. The administrator is the main user of this web application and he can add employee details, vacancy details, HR details, etc.

Our HRMS project is being developed in such a way that it can control and manage employee's data in a more appropriate and neat way. With the help of our project, we will have better software to manage personal data, control mechanisms to authorize and authenticate for the employees' entry.

FEATURES :

1. Benefits administration
2. Self service(management)
3. Self service(employee)
4. Automated recruitment
5. Automated data gathering
6. Payroll integration
7. Employee Scheduling and many more

ADVANTAGES:

1. Many organizations manage their workforce without HRMS software assistance. Still, with a robust HRMS, everyone with access within the company can perform task management and access self-services.
2. Executives or supervisors can assign new tasks, manage attendance of their staff, and track the location in their handy mobile application.
3. The system allows organizing company policies, improves employee and management collaboration, and increases its overall revenue.
4. HRMS can automate entire HR operations, such as:
   - Managing payroll;
   - Recruitment and hiring;
   - Collecting, storing, and reaching the required data;
   - Calculation of wages and other payments;
   - Performance evaluation;
   - Managing employees' training;
   - Bonuses and compensation administration;
   - Keeping statistics on HR and recruiting processes: staff turnover, number of closed and open vacancies, recruiting funnel;
   - Employee self-service;
   - Automatic posting of vacancies on job portals using integration;
5. An inclusive HRMS is like a well-oiled machine where all the components work together to keep it moving forward, carrying out its function.

DISADVANTAGES:

1.Privacy Concerns:
One of the chief disadvantages of HR management systems is the increased capacity for breaches in the privacy of the employees, management and business officials.
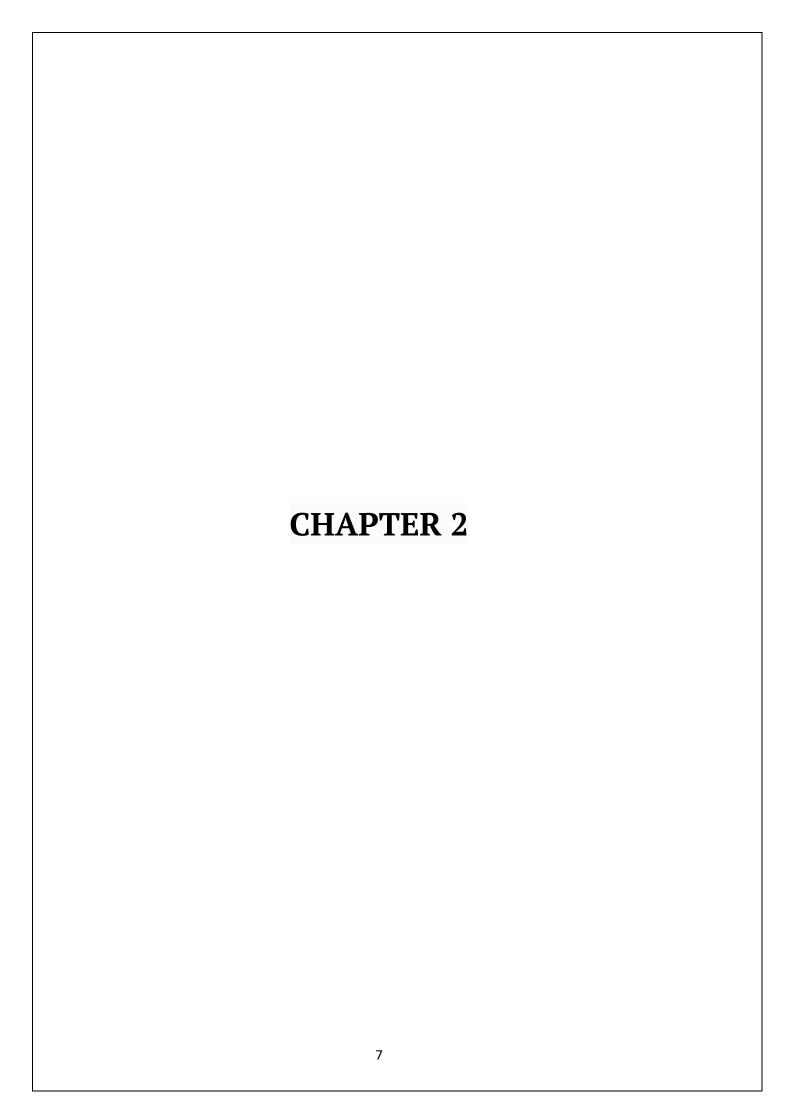
2.Security:
On par with ruptures in privacy protection are fissures in the security system protecting your HR management software.

3.User Error:
Inherent in the use of HR management systems is the means for inaccuracy. Because computers and their associated programs are only as effective as their human users, data entry errors can and do occur.

4.Employee Assessment:

HR management systems can be effective in selecting employees for certain positions or for advancement, based on performance scores and other information.

CHAPTER 2

OBJECTIVE:

The purpose for this project is to design a database system for Human Resource and Payroll Management System together with the required queries.

A few factors that direct us to develop a new system are given below:

1) Faster System

2) Accuracy

3) Reliability

4) Informative

5) Convenience

SCOPE OF THE PROJECT :

This document covers the whole definition of the HR Management System (HRMS) project. It basically includes the requirements for managing the personal data, controlling authentication and authorization mechanism, and evaluating employees' performance. After creating the new HRMS we have to accomplish data migration from their existing system to our new one. More specifically, our HRMS (HR Management System) controls and manages the personal database such that any user with different role types as manager, admin, employee, and human resource will be able to manipulate their personal data. In addition to manipulating the personal data, our HRMS will provide authentication and authorization mechanisms. Every user with any role type can be able to login to the system with his/her username and password.

RELATIONAL DATABASE DESIGN:

1. **EMPLOYEE:**
   Emp_id, name, contact, Join_date, designation, dept_id, Pay id, ded_id, att_id

2. **DEPARTMENT:**
   Dep_id, dep_name, no_emp, dept_head

3. **PAYMENT:**
   Pay_id, ba, da, hra, gross_pay, pay_date

4. **DEDUCTIONS:**
   de_id,type,deduction_amount,s_month

5. **PROJECT:**
   project_id,title,document,start_date,end_date,project_status

6. **INTERVIEW:**
   interview_id,selection_round,interview_date,venue,candidate_no,int_status

7. **APPLICANTS:**
   app_id,name,candidate_no,qualification,app_date,app_status

8. **ATTENDANCE:**
   att_id,date,login_time,logout_time,total_hour

9. **VACANCIES:**
   vac_id,criteria,qualification,dept_id

REQUIREMENT ANALYSIS:

1.EMPLOYEE:
In the Employee table we have emp_id, name, contact, join_date, designation dept_id, pay_id and att_id.Out of them emp_id is taken as primary key because every employee will have a unique id with which we can identify a particular applicant. Here the information about employee's departments, payment, deduction and attendance will be saved.

2.DEPARTMENT:
In the Departments table we have dept_id, dept_name, no_emp, dept_head.Out of them dept_id is taken as primary key because every department will have a unique id with which we can identify a particular applicant. In this table all the departments names with the number of employees working in them along with department head names will be saved.

3.PAYMENT:

In the Payment table we have pay_id, ba, da, hra, gross_pay and pay_date.Out of them pay_id is taken as primary key because every payment will have a unique id with which we can identify a particular applicant. Here the information about employee's each payment with the breakdown of gross pay and payment date will be saved.

4.DEDUCTIONS:

In the deductions table we have de_id, type, deduction_amount, s_month. Out of them de_id is taken as the primary key because every employee will have a unique id with which we can identify a particular employee.

5.PROJECT:

In the project table we have project_id, title, document, start_date, end_date, project_status. Out of them, project_id is taken as the primary key because every project will have a unique id with which we can identify a particular project.

6.INTERVIEW:

In the interview table we have interview_id, selection_round, interview_date, venue. Out of them interview_id is taken as the primary key because every applicant will have a unique id with which we can identify a particular applicant.

7.APPLICANTS:

In the applicants table we have app_id,name,candidate_no,qualification,app_date,app_status.Out of them app_id is taken as primary key because every applicant will have a unique id with which we can identify a particular applicant.
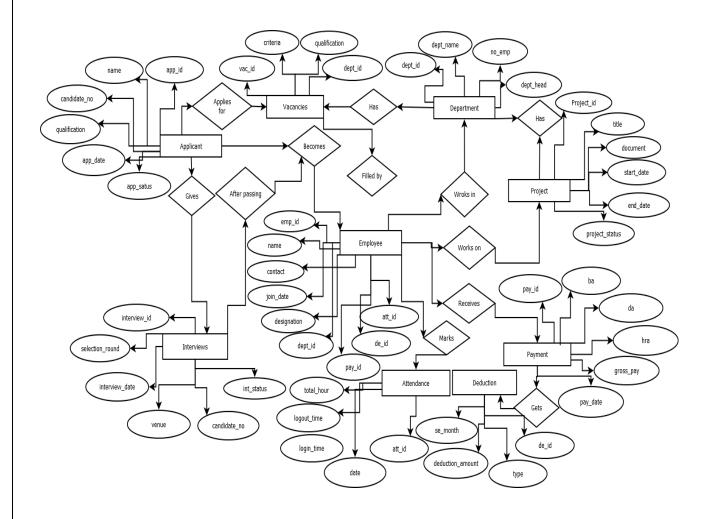
8.ATTENDANCE:

In the attendance table we have five attributes: att_id,date,login_time,logout_time,total_hour.Out of them att-id is taken as primary key because every employee will have a unique id with which we can identify the attendance.

9.VACANCIES:

In the vacancies table we have four attributes:
vac_id,criteria,qualification,dept_id.Out of them vac_id is taken as primary key because every department will have a unique id with which we can identify the vacancies in the department.

CHAPTER 3

ER DIAGRAM:

SCHEMA TABLES:

1.EMPLOYEE:
Primary key: emp_id

emp_id	name	join_date	contact	designation	dept_id	pay_id	de_id	att_id

2.DEPARTMENT:
Primary key: dept_id

dept_id	dept_name	no_emp	dept_head

3.PAYMENT:
Primary key: pay_id

pay_id	ba	da	hra	gross_pay	pay_date

4.DEDUCTIONS:
Primary key: de_id

de_id	type	deduction_amount	s_month

5.PROJECT:
Primary key: project_id

project_id	title	document	start_date	end_date	project_status

6.INTERVIEW:
Primary key: interview_id

interview_id	selection_rund	interview_date	venue	candidate_no	int_status

7.APPLICANTS:

Primary key: app_id

app_id	name	candidate_no	qualification	app_date	app_status

8.ATTENDANCE:

Primary key: att_id

att_id	date	login_time	logout_time	total_hour

9.VACANCIES:

Primary key: vac_id

vac_id	criteria	qualification	dept_id

CHAPTER 4

SQL COMMANDS:

1. Create the Database

Query: Create a Database name 'hrms'

Syntax: create database <database name>;

```
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (13.4)
WARNING: Console code page (437) differs from Windows code page (1252)
        8-bit characters might not work correctly. See psql reference
        page "Notes for Windows users" for details.
Type "help" for help.

postgres=# create database hrms;
CREATE DATABASE
postgres=# \c hrms
You are now connected to database "hrms" as user "postgres".
```

2. Create Tables

Syntax: create table <table name>(column1 datatype, column2 datatype,.......,column n datatype);

Query: Create a table name 'Employee' with attributes Employee id, Name, Contact, Join date, Designation, Department id, Pay id, Deduction id, Attendance id

```
hrms=# create table Employee (Emp_id integer, Name varchar(50), Contact integer, Join_date date,
Designation varchar(50), Dept_id integer, Pay_id integer, DE_id integer, Att_id integer)
hrms-#
hrms-# ;
CREATE TABLE
hrms=# select*from Employee
hrms-# ;
 emp_id | name | contact | join_date | designation | dept_id | pay_id | de_id | att_id
--------+------+---------+-----------+-------------+---------+--------+-------+--------
(0 rows)
```

Query: Create a table 'Department' with attributes Department id, Department name, Number of employees and Department head name

```
hrms=# create table Department (Dept_id integer, Dept_name varchar(50), no_emp integer,
 Dept_Head varchar(50));
CREATE TABLE
hrms=# select*from Department;
 dept_id | dept_name | no_emp | dept_head
---------+-----------+--------+-----------
(0 rows)
```

Query: Create table 'Payment' with attributes Payment id, Basic pay, DA, HRA, Gross pay and payment date

```
hrms=# create table Payment (Pay_id integer, BA integer, DA integer, HRA integer, Gross_pay integer,
Pay_date date);
CREATE TABLE
hrms=# select*from Payment;
 pay_id | ba | da | hra | gross_pay | pay_date
--------+----+----+-----+-----------+----------
(0 rows)
```

Query: Create table 'Deductions' with attributes de_id, type, deduction_amount, s_month.

```
hrms=# alter table deductions
hrms-# rename column decution_amount to deduction_amount;
ALTER TABLE
hrms=# select * from deductions;
 de_id | type | deduction_amount | s_month
-------+------+------------------+---------
(0 rows)
```

Query: Create table 'Project' with attributes project_id, title, document, start_date, end_date, project_status.

```
hrms=# create table project(project_id integer primary key, title varchar(50), document varchar(50), start_date date, end_date date,project_status varchar(50));
CREATE TABLE
hrms=# select * from project;
 project_id | title | document | start_date | end_date | project_status
------------+-------+----------+------------+----------+----------------
(0 rows)
```

Query: Create table 'Interview' with attributes interview_id, selection_round, interview_date, venue.

```
hrms=# create table interview(interview_id integer primary key, selection_round integer, interview_date date, venue varchar(50));
CREATE TABLE
hrms=# select * from interview;
 interview_id | selection_round | interview_date | venue
--------------+-----------------+----------------+-------
(0 rows)
```

Query: Create table 'Applicants' with attributes app_id,name,candidate_no,qualification,app_date,app_status

```
hrms=# create table applicants(app_id integer,name varchar(50), candidate_no integer,qualification varchar(50),app_date date,app_status varchar(50));
CREATE TABLE
hrms=# select*from applicants;
 app_id | name | candidate_no | qualification | app_date | app_status
--------+------+--------------+---------------+----------+------------
(0 rows)
```

Query: Create table 'Attendance' with attributes att_id,date,login_time,logout_time,total_hour.

```
hrms=# create table attendance(att_id integer,date date,login_time time,logout_time time,total_hours time);
CREATE TABLE
hrms=# select*from attendance;
 att_id | date | login_time | logout_time | total_hours
--------+------+------------+-------------+-------------
(0 rows)
```

Query: Create table 'Vacancies' with attributes vac_id,criteria,qualification,dept_id.

```
hrms=# create table vacancies(vac_id integer,criteria varchar(50),qualification varchar(50),dept_id integer);
CREATE TABLE
hrms=# select*from vacancies;
 vac_id | criteria | qualification | dept_id
--------+----------+---------------+---------
(0 rows)
```

3. Insert values into the tables:

Syntax: insert into <table name> values (column1, column2,........, column n);

Query: Insert Employee details in 'Employee' table

```
hrms=# insert into Employee values ('1001','Santosh Joshi','9183849','21-01-2001','Acounting Head','101',
'20001','56987','12345'),('1002','Mitali Mishra','7291910','05-03-2011','Project Manager','102','20010','
15976','56789'),('1010','Jay Mittal','8478930','28-05-2008','Graphic Designer','103','21001','84267','678
90'),('1213','Rupali Kulkarni','958577','17-10-2003','Accountant','101','23450','79315','23456'),('2901',
'Ramesh Tripathi','7299390','26-09-2005','Programmer','104','21012','23789','12678'),('1345','Sonali Sikk
a','894949','06-02-2004','Marketing_Head','105','29108','35794','34567');
INSERT 0 6
hrms=# select*from Employee;
 emp_id |      name       | contact | join_date  |   designation    | dept_id | pay_id | de_id | att_id
--------+-----------------+---------+------------+------------------+---------+--------+-------+--------
   1001 | Santosh Joshi   | 9183849 | 2001-01-21 | Acounting Head   |     101 |  20001 | 56987 |  12345
   1002 | Mitali Mishra   | 7291910 | 2011-03-05 | Project Manager  |     102 |  20010 | 15976 |  56789
   1010 | Jay Mittal      | 8478930 | 2008-05-28 | Graphic Designer |     103 |  21001 | 84267 |  67890
   1213 | Rupali Kulkarni |  958577 | 2003-10-17 | Accountant       |     101 |  23450 | 79315 |  23456
   2901 | Ramesh Tripathi | 7299390 | 2005-09-26 | Programmer       |     104 |  21012 | 23789 |  12678
   1345 | Sonali Sikka    |  894949 | 2004-02-06 | Marketing_Head   |     105 |  29108 | 35794 |  34567
(6 rows)
```

Query: Insert details of existing departments in 'Department' table

```
hrms=# insert into Department values ('101','Acounting','20','Santosh Joshi'),('102','Management
','30','Rutuja Sathe'),('103','Designing','30','Samir Agrawal'),('104','Programming','50','Seema
 Mishra'),('105','Marketing','40','Sonali Sikka'),('106','Customer Service','10','Mehram Jha');
INSERT 0 6
hrms=# select*from Department;
 dept_id |    dept_name     | no_emp |   dept_head
---------+------------------+--------+---------------
     101 | Acounting        |     20 | Santosh Joshi
     102 | Management       |     30 | Rutuja Sathe
     103 | Designing        |     30 | Samir Agrawal
     104 | Programming      |     50 | Seema Mishra
     105 | Marketing        |     40 | Sonali Sikka
     106 | Customer Service |     10 | Mehram Jha
(6 rows)
```

Query: Insert Details of payment of employees into 'Payment' table

```
hrms=# insert into Payment values ('20001','50000','10000','15000','75000','10-12-2021')
,('20010','65000','12000','15000','92000','10-12-2021'),('21001','40000','10000','10000'
,'60000','10-12-2021'),('23450','30000','10000','10000','50000','10-12-2021'),('21012','
70000','15000','15000','100000','10-12-2021'),('29108','50000','10000','10000','70000','
10-12-2021');
INSERT 0 6
hrms=# select*from Payment;
 pay_id |  ba   |  da   |  hra  | gross_pay |  pay_date
--------+-------+-------+-------+-----------+------------
  20001 | 50000 | 10000 | 15000 |     75000 | 2021-12-10
  20010 | 65000 | 12000 | 15000 |     92000 | 2021-12-10
  21001 | 40000 | 10000 | 10000 |     60000 | 2021-12-10
  23450 | 30000 | 10000 | 10000 |     50000 | 2021-12-10
  21012 | 70000 | 15000 | 15000 |    100000 | 2021-12-10
  29108 | 50000 | 10000 | 10000 |     70000 | 2021-12-10
(6 rows)
```

Query: Insert Details of payment deduction of employees into 'Deductions' table

```
hrms=# insert into deductions(de_id, type, deduction_amount, s_month) values ('35794', 'Job-Related Expenses', '7000', 'August'),('45589', 'Retirement Plans', '35000', 'June'),
('57423', 'Life Insurance Premiums', '12000', 'July'), ('92275', 'Health Insurance Premiums', '24000', 'June'), ('18379', 'Life Insurance Premiums', '15000', 'May');
INSERT 0 5
hrms=# select * from deductions;
 de_id |          type           | deduction_amount | s_month
-------+-------------------------+------------------+---------
 56987 | Health Insurance Premiums |          25000 | August
 15976 | Retirement Plans        |          30000 | June
 84267 | Life Insurance Premiums |          10000 | July
 79315 | Job-Related Expenses    |           5000 | June
 23789 | Health Insurance Premiums |          15000 | May
 35794 | Job-Related Expenses    |           7000 | August
 45589 | Retirement Plans        |          35000 | June
 57423 | Life Insurance Premiums |          12000 | July
 92275 | Health Insurance Premiums |          24000 | June
 18379 | Life Insurance Premiums |          15000 | May
(10 rows)
```

Query: Insert Details of project into 'Project' table

```
hrms=# select * from project;
 project_id |         title            |       document        | start_date |  end_date  | project_status
------------+--------------------------+-----------------------+------------+------------+----------------
       1234 | Cloud Database           | Datasheet             | 2021-01-15 | 2021-05-25 | Completed
       5678 | Payment Gateway          | Accounts              | 2021-08-10 | 2021-10-11 | Approved
       9101 | Website Design Templates | Scripts               | 2021-10-15 | 2022-01-30 | Working
       1121 | Data Import Export Plugin | Plugin Data          | 2021-01-10 | 2021-10-20 | Pending
       3141 | Research & Development   | Thesis/Reports        | 2021-06-15 | 2021-09-25 | Finalised
       5161 | E-Learning Integration   | SRS                   | 2021-12-02 | 2022-05-25 | Started
       7181 | Portal System            | Research              | 2021-03-20 | 2021-08-30 | Completed
       9202 | User Interface Design    | Design                | 2021-01-10 | 2021-04-20 | Completed
       1222 | Wellness App             | Android Study Material | 2021-04-15 | 2021-07-25 | Approved
       3242 | Unit Testing             | Test Tools            | 2021-02-12 | 2021-07-10 | Approved
(10 rows)
```

Query: Insert Details of interviews of applicants into 'Interview' table

```
hrms=# insert into interview(interview_id, selection_round, interview_date, venue) values ('00677', '2', '2021-11-25', 'Pune Office'),('00794', '3', '2021-12-12', 'Mumbai Office'),
('00841', '2', '2021-10-15', 'Pune Office'), ('00955', '3', '2021-04-14', 'Chennai Office'), ('00623', '1', '2021-08-15', 'Chennai Office');
INSERT 0 5
hrms=# select * from interview;
 interview_id | selection_round | interview_date |     venue
--------------+-----------------+----------------+----------------
          248 |               1 | 2021-05-12     | Pune Office
          481 |               2 | 2021-06-12     | Mumbai Office
          396 |               1 | 2021-10-10     | Pune Office
          567 |               3 | 2021-02-12     | Kolkata Office
          124 |               2 | 2021-08-20     | Chennai Office
          677 |               2 | 2021-11-25     | Pune Office
          794 |               3 | 2021-12-12     | Mumbai Office
          841 |               2 | 2021-10-15     | Pune Office
          955 |               3 | 2021-04-14     | Chennai Office
          623 |               1 | 2021-08-15     | Chennai Office
(10 rows)
```

Query: Insert Details of applicants into 'Applicants' table

```
hrms=# insert into applicants
hrms-# values('12345','jonas','1001','btech','1/11/2021','active');
INSERT 0 1
hrms=# insert into applicants
hrms-# values('56789','scarlet','1002','mtech','5/11/2021','offer');
INSERT 0 1
hrms=# insert into applicants
hrms-# values('67890','frank','1003','mba','7/11/2021','first interview');
INSERT 0 1
hrms=# insert into applicants
hrms-# values('23456','marker','1004','btech','4/11/2021','second interview');
INSERT 0 1
hrms=# insert into applicants
hrms-# values('12678','blaze','1005','mtech','2/11/2021','accepted');
INSERT 0 1
hrms=# insert into applicants
hrms-# values('34567','clare','1006','btech','6/11/2021','active');
INSERT 0 1
```

Query: Insert Details of attendance of the employees into 'Attendance' table

```
hrms=# insert into attendance
hrms-# values('12345','1/11/2021','9:30','10:30','1');
ERROR:  invalid input syntax for type time: "1"
LINE 2: values('12345','1/11/2021','9:30','10:30','1');
                                                   ^
hrms=# insert into attendance
hrms-# values('12345','1/11/2021','9:30','10:30','1:00');
INSERT 0 1
hrms=# insert into attendance
hrms-# values('56789','5/11/2021','2:40','2:50','00:10');
INSERT 0 1
hrms=# insert into attendance
hrms-# values('67890','7/11/2021','11:30','11:50','00:20');
INSERT 0 1
hrms=# insert into attendance
hrms-# values('23456','4/11/2021','12:30','1:30','1:00');
INSERT 0 1
hrms=# insert into attendance
hrms-# values('12678','2/11/2021','3:40','4:10','00:30');
INSERT 0 1
hrms=# insert into attendance
hrms-# values('34567','6/11/2021','2:00','3:00','1:00');
INSERT 0 1
```

Query: Insert Details of vacancies into 'Vacancies' table

```
INSERT 0 1
hrms=# insert into vacancies
hrms-# values('4444','leadership','btech','11112');
INSERT 0 1
hrms=# insert into vacancies
hrms-# values('5555','communication skills','btech','11113');
INSERT 0 1
hrms=# insert into vacancies
hrms-# values('6666','gpa','btech','11114');
INSERT 0 1
hrms=# insert into vacancies
hrms-# values('7777','training experience','btech','11114');
INSERT 0 1
hrms=# insert into vacancies
hrms-# values('8888','risk evaluation ability','btech','11115');
INSERT 0 1
```

4. View all the contents of tables:

Syntax: SELECT*FROM <table ame>;

Applicants:

```
hrms=# select*from applicants;
 app_id |  name  | candidate_no | qualification |  app_date  |    app_status
--------+--------+--------------+---------------+------------+------------------
  12345 | jonas  |         1001 | btech         | 2021-11-01 | active
  56789 | scarlet|         1002 | mtech         | 2021-11-05 | offer
  67890 | frank  |         1003 | mba           | 2021-11-07 | first interview
  23456 | marker |         1004 | btech         | 2021-11-04 | second interview
  12678 | blaze  |         1005 | mtech         | 2021-11-02 | accepted
  34567 | clare  |         1006 | btech         | 2021-11-06 | active
(6 rows)
```

Attendance:

```
hrms=# select*from attendance;
 att_id |    date    | login_time | logout_time | total_hours
--------+------------+------------+-------------+-------------
  12345 | 2021-11-01 | 09:30:00   | 10:30:00    | 01:00:00
  56789 | 2021-11-05 | 02:40:00   | 02:50:00    | 00:10:00
  67890 | 2021-11-07 | 11:30:00   | 11:50:00    | 00:20:00
  23456 | 2021-11-04 | 12:30:00   | 01:30:00    | 01:00:00
  12678 | 2021-11-02 | 03:40:00   | 04:10:00    | 00:30:00
  34567 | 2021-11-06 | 02:00:00   | 03:00:00    | 01:00:00
(6 rows)
```

Vacancies:

```
hrms=# select*from vacancies;
 vac_id |        criteria        | qualification | dept_id
--------+------------------------+---------------+---------
   3333 | Work experience        | btech         |   11111
   4444 | leadership             | btech         |   11112
   5555 | communication skills   | btech         |   11113
   6666 | gpa                    | btech         |   11114
   7777 | training experience    | btech         |   11114
   8888 | risk evaluation ability| btech         |   11115
(6 rows)
```

5. Drop

Syntax: drop table <table name>;

Query: Delete table 'Department' from the database

```
hrms=# select*from Department;
 dept_id | dept_name | no_emp | pro_id | dept_head
---------+-----------+--------+--------+-----------
(0 rows)



hrms=# drop table Department;
DROP TABLE
hrms=# select*from Department;
ERROR:  relation "department" does not exist
LINE 1: select*from Department;
                    ^
```

6. Select

Syntax: SELECT <attribute> FROM <table name> WHERE <condition>;

Query: Find name of all employees who work in Accounting department

```
hrms=# SELECT*FROM Employee;
 emp_id |      name       | contact | join_date  |   designation    | dept_id | pay_id | de_id | att_id
--------+-----------------+---------+------------+------------------+---------+--------+-------+--------
   1001 | Santosh Joshi   | 9183849 | 2001-01-21 | Acounting Head   |     101 |  20001 | 56987 |  12345
   1002 | Mitali Mishra   | 7291910 | 2011-03-05 | Project Manager  |     102 |  20010 | 15976 |  56789
   1010 | Jay Mittal      | 8478930 | 2008-05-28 | Graphic Designer |     103 |  21001 | 84267 |  67890
   1213 | Rupali Kulkarni |  958577 | 2003-10-17 | Accountant       |     101 |  23450 | 79315 |  23456
   2901 | Ramesh Tripathi | 7299390 | 2005-09-26 | Programmer       |     104 |  21012 | 23789 |  12678
   1345 | Sonali Sikka    |  894949 | 2004-02-06 | Marketing_Head   |     105 |  29108 | 35794 |  34567
(6 rows)


hrms=# SELECT name FROM Employee WHERE dept_id='101';
      name
----------------
 Santosh Joshi
 Rupali Kulkarni
(2 rows)
```

7. Update

Syntax: UPDATE <table name> SET <column n = value n> WHERE <condition>;

Query: Increase Gross pay of all employees by 'Rs. 5000'; where dearness allowance is 'Rs. 10000'

```
hrms=# UPDATE Payment SET gross_pay=gross_pay+5000 WHERE da='10000';
UPDATE 4
hrms=# SELECT*FROM Payment;
 pay_id |  ba   |  da   |  hra  | gross_pay |  pay_date
--------+-------+-------+-------+-----------+------------
  20010 | 65000 | 12000 | 15000 |     92000 | 2021-12-10
  21012 | 70000 | 15000 | 15000 |    100000 | 2021-12-10
  20001 | 50000 | 10000 | 15000 |     80000 | 2021-12-10
  21001 | 40000 | 10000 | 10000 |     65000 | 2021-12-10
  23450 | 30000 | 10000 | 10000 |     55000 | 2021-12-10
  29108 | 50000 | 10000 | 10000 |     75000 | 2021-12-10
(6 rows)
```

8. Alter

Syntax: ALTER TABLE <table name> ADD/DELETE COLUMN <column name datatype>;

Query: Add column deduction id into Payment table

```
hrms=# ALTER TABLE Payment ADD COLUMN de_id integer;
ALTER TABLE
hrms=# select*from Payment;
 pay_id |  ba   |  da   |  hra  | gross_pay |  pay_date  | de_id
--------+-------+-------+-------+-----------+------------+-------
  20010 | 65000 | 12000 | 15000 |     92000 | 2021-12-10 |
  21012 | 70000 | 15000 | 15000 |    100000 | 2021-12-10 |
  20001 | 50000 | 10000 | 15000 |     80000 | 2021-12-10 |
  21001 | 40000 | 10000 | 10000 |     65000 | 2021-12-10 |
  23450 | 30000 | 10000 | 10000 |     55000 | 2021-12-10 |
  29108 | 50000 | 10000 | 10000 |     75000 | 2021-12-10 |
(6 rows)
```

Query: Delete column de_id from Employee table

```
hrms=# ALTER TABLE Employee DROP COLUMN de_id;
ALTER TABLE
hrms=# select*from Employee;
 emp_id |      name       | contact | join_date  |   designation    | dept_id | pay_id | att_id
--------+-----------------+---------+------------+------------------+---------+--------+-------
   1001 | Santosh Joshi   | 9183849 | 2001-01-21 | Acounting Head   |     101 |  20001 | 12345
   1002 | Mitali Mishra   | 7291910 | 2011-03-05 | Project Manager  |     102 |  20010 | 56789
   1010 | Jay Mittal      | 8478930 | 2008-05-28 | Graphic Designer |     103 |  21001 | 67890
   1213 | Rupali Kulkarni |  958577 | 2003-10-17 | Accountant       |     101 |  23450 | 23456
   2901 | Ramesh Tripathi | 7299390 | 2005-09-26 | Programmer       |     104 |  21012 | 12678
   1345 | Sonali Sikka    |  894949 | 2004-02-06 | Marketing_Head   |     105 |  29108 | 34567
(6 rows)
```

9. Delete

Syntax: DELETE FROM <table name> WHERE <condition>;

Query: Delete 'Customer Service' department from 'Department' table

```
hrms=# DELETE FROM Department WHERE dept_name='Customer Service';
DELETE 1
hrms=# select*from Department;
 dept_id |  dept_name  | no_emp |   dept_head
---------+-------------+--------+---------------
     101 | Acounting   |     20 | Santosh Joshi
     102 | Management  |     30 | Rutuja Sathe
     103 | Designing   |     30 | Samir Agrawal
     104 | Programming |     50 | Seema Mishra
     105 | Marketing   |     40 | Sonali Sikka
(5 rows)
```

10.   AND

Syntax: <condition 1> AND <condition 2>

Query: Find name of the employee who is from 'Accounting' department and has salary 'Rs.1,00,000'

```
rms=# select*from Employee;
emp_id |      name       | contact | join_date  |  designation    | dept_id | pay_id | att_id
-------+-----------------+---------+------------+-----------------+---------+--------+--------
  1001 | Santosh Joshi   | 9183849 | 2001-01-21 | Acounting Head  |     101 |  20001 |  12345
  1002 | Mitali Mishra   | 7291910 | 2011-03-05 | Project Manager |     102 |  20010 |  56789
  1010 | Jay Mittal      | 8478930 | 2008-05-28 | Graphic Designer|     103 |  21001 |  67890
  1213 | Rupali Kulkarni |  958577 | 2003-10-17 | Accountant      |     101 |  23450 |  23456
  2901 | Ramesh Tripathi | 7299390 | 2005-09-26 | Programmer      |     104 |  21012 |  12678
  1345 | Sonali Sikka    |  894949 | 2004-02-06 | Marketing_Head  |     105 |  29108 |  34567
6 rows)

rms=# SELECT name FROM Employee WHERE dept_id='104' AND pay_id='21012';
     name
---------------
Ramesh Tripathi
1 row)
```

11.   OR

Syntax: <condition 1> OR <condition 2>

Query: Find the name of the departments where number of employees is either 30 or 50

```
hrms=# select*from department;
 dept_id |  dept_name  | no_emp |   dept_head
---------+-------------+--------+---------------
     101 | Acounting   |     20 | Santosh Joshi
     102 | Management  |     30 | Rutuja Sathe
     103 | Designing   |     30 | Samir Agrawal
     104 | Programming |     50 | Seema Mishra
     105 | Marketing   |     40 | Sonali Sikka
(5 rows)


hrms=# SELECT dept_name FROM Department WHERE no_emp='30' OR no_emp='50';
  dept_name
-------------
 Management
 Designing
 Programming
(3 rows)
```

12.   IN

Syntax: value IN (value 1, value 2.....);

Queries: Find name of all employees working in accounting, management and designing department

```
hrms=# select*from Employee;
 emp_id |      name       | contact | join_date  |   designation    | dept_id | pay_id | att_id
--------+-----------------+---------+------------+------------------+---------+--------+--------
   1001 | Santosh Joshi   | 9183849 | 2001-01-21 | Acounting Head   |     101 |  20001 |  12345
   1002 | Mitali Mishra   | 7291910 | 2011-03-05 | Project Manager  |     102 |  20010 |  56789
   1010 | Jay Mittal      | 8478930 | 2008-05-28 | Graphic Designer |     103 |  21001 |  67890
   1213 | Rupali Kulkarni |  958577 | 2003-10-17 | Accountant       |     101 |  23450 |  23456
   2901 | Ramesh Tripathi | 7299390 | 2005-09-26 | Programmer       |     104 |  21012 |  12678
   1345 | Sonali Sikka    |  894949 | 2004-02-06 | Marketing_Head   |     105 |  29108 |  34567
(6 rows)


hrms=# SELECT name FROM Employee WHERE dept_id IN (101, 102, 203)
hrms-# ;
     name
----------------
 Santosh Joshi
 Mitali Mishra
 Rupali Kulkarni
(3 rows)
```

13.   NOT-IN

Syntax: value NOT IN (value 1, value 2,....);

Query: Find the name of the departments Which Does not have 20,40 and 50 employees

```
hrms=# select*from Department;
 dept_id |  dept_name  | no_emp |   dept_head
---------+-------------+--------+---------------
     101 | Acounting   |     20 | Santosh Joshi
     102 | Management  |     30 | Rutuja Sathe
     103 | Designing   |     30 | Samir Agrawal
     104 | Programming |     50 | Seema Mishra
     105 | Marketing   |     40 | Sonali Sikka
(5 rows)


hrms=# SELECT dept_name FROM Department WHERE no_emp NOT IN (20, 40, 50);
 dept_name
------------
 Management
 Designing
(2 rows)
```

14.   BETWEEN

Syntax: value BETWEEN low AND high;

Query: Find the Projects which have been Completed in the First Half of the Year 2021.

```
hrms=# SELECT project_id, title, project_status FROM project WHERE end_date BETWEEN '2021-01-01' AND '2021-06-30'
 AND project_status='Completed';
 project_id |        title         | project_status
------------+----------------------+----------------
       1234 | Cloud Database       | Completed
       9202 | User Interface Design | Completed
(2 rows)
```

15.   NOT BETWEEN

Syntax: value NOT BETWEEN low AND high;

Query: Find the Deductions not in the range of 5000 to 15000.

```
hrms=# SELECT de_id, deduction_amount FROM deductions WHERE deduction_amount NOT BETWEEN 5000 AND 15000;
 de_id | deduction_amount
-------+------------------
 56987 |            25000
 15976 |            30000
 45589 |            35000
 92275 |            24000
(4 rows)
```

16.   LIKE

Syntax: Value LIKE pattern

Query: Find the Interview details of which took place in the Pune Office.

```
hrms=# SELECT interview_id, interview_date FROM interview WHERE venue LIKE 'Pune%';
 interview_id | interview_date
--------------+----------------
          248 | 2021-05-12
          396 | 2021-10-10
          677 | 2021-11-25
          841 | 2021-10-15
(4 rows)
```

17.   NOT LIKE

Syntax: Value NOT LIKE pattern

Query: Find the Projects which have not yet been Completed.

```
hrms=# SELECT title, project_status FROM project WHERE project_status NOT LIKE 'C%';
           title            | project_status
----------------------------+----------------
 Payment Gateway            | Approved
 Website Design Templates   | Working
 Data Import Export Plugin  | Pending
 Research & Development     | Finalised
 E-Learning Integration     | Started
 Wellness App               | Approved
 Unit Testing               | Approved
(7 rows)
```

28

18.   SUM

Syntax: SUM(DISTINCT expression)

Query: Find the Total Deductions of all employees.

```
hrms=# SELECT SUM(deduction_amount) AS Total_Deduction FROM deductions;
 total_deduction
----------------
          178000
(1 row)
```

19.   COUNT

Syntax: SELECT COUNT(*) FROM table_name WHERE condition;

Query: Find the number of employees who have the Deduction Type as Life Insurance Premium.

```
hrms=# SELECT COUNT(*) FROM deductions WHERE type LIKE 'Life%';
 count
-------
     3
(1 row)
```

20.   MAX

Syntax: MAX(expression);

Query: Find the Project with the longest duration of work.

```
hrms=# SELECT MAX(end_date - start_date) FROM project;
 max
-----
 283
(1 row)
```

```
hrms=# SELECT project_id, title, AGE(end_date, start_date) AS maximum_duration FROM project ORDER BY
maximum_duration DESC LIMIT 1;
 project_id |           title           | maximum_duration
------------+---------------------------+------------------
       1121 | Data Import Export Plugin | 9 mons 10 days
(1 row)
```

21. MIN

Syntax: MIN(expression);

Query: Find the Least Deduction Amount.

```
hrms=# SELECT MIN(deduction_amount) FROM deductions;
 min
------
 5000
(1 row)
```

22. AVERAGE

Syntax: AVG(column)

Query: Find the Average Deductions Amount.

```
hrms=# SELECT AVG(deduction_amount) FROM deductions;
        avg
--------------------
 17800.000000000000
(1 row)
```

23. GROUP BY

Syntax :
SELECT column_name(s)
FROM table_name
WHERE condition
GROUP BY column_name(s)
ORDER BY column_name(s);

Query : Find the total logout time from table attendance

```
hrms=# select date,sum(logout_time) from attendance group by date;
    date    |   sum
------------+----------
 2021-11-05 | 02:50:00
 2021-11-02 | 04:10:00
 2021-11-07 | 11:50:00
 2021-11-01 | 10:30:00
 2021-11-06 | 03:00:00
 2021-11-04 | 01:30:00
(6 rows)
```

24.   ORDER BY

Syntax :
SELECT column1, column2, ...
FROM table_name
ORDER BY column1, column2, ... ASC|DESC;

Query : List the name of all applicants in descending order.

```
hrms=# select*from applicants order by name DESC;
 app_id |  name   | candidate_no | qualification |  app_date  |    app_status
--------+---------+--------------+---------------+------------+------------------
  56789 | scarlet |         1002 | mtech         | 2021-11-05 | offer
  23456 | marker  |         1004 | btech         | 2021-11-04 | second interview
  12345 | jonas   |         1001 | btech         | 2021-11-01 | active
  67890 | frank   |         1003 | mba           | 2021-11-07 | first interview
  34567 | clare   |         1006 | btech         | 2021-11-06 | active
  12678 | blaze   |         1005 | mtech         | 2021-11-02 | accepted
(6 rows)
```

25.   CROSS JOIN

Syntax :
SELECT column_name(s)
FROM table1
CROSS JOIN table2;

Query : To obtain all the paired combinations from the applicants and attendance table.

```
hrms=# select*from applicants CROSS JOIN attendance;
 app_id |  name   | candidate_no | qualification | app_date   |   app_status     | att_id |    date    | login_time | logout_time | total_hours
--------+---------+--------------+---------------+------------+------------------+--------+------------+------------+-------------+------------
  12345 | jonas   |         1001 | btech         | 2021-11-01 | active           |  12345 | 2021-11-01 | 09:30:00   | 10:30:00    | 01:00:00
  56789 | scarlet |         1002 | mtech         | 2021-11-05 | offer            |  12345 | 2021-11-01 | 09:30:00   | 10:30:00    | 01:00:00
  67890 | frank   |         1003 | mba           | 2021-11-07 | first interview  |  12345 | 2021-11-01 | 09:30:00   | 10:30:00    | 01:00:00
  23456 | marker  |         1004 | btech         | 2021-11-04 | second interview |  12345 | 2021-11-01 | 09:30:00   | 10:30:00    | 01:00:00
  12678 | blaze   |         1005 | mtech         | 2021-11-02 | accepted         |  12345 | 2021-11-01 | 09:30:00   | 10:30:00    | 01:00:00
  34567 | clare   |         1006 | btech         | 2021-11-06 | active           |  12345 | 2021-11-01 | 09:30:00   | 10:30:00    | 01:00:00
  12345 | jonas   |         1001 | btech         | 2021-11-01 | active           |  56789 | 2021-11-05 | 02:40:00   | 02:50:00    | 00:10:00
  56789 | scarlet |         1002 | mtech         | 2021-11-05 | offer            |  56789 | 2021-11-05 | 02:40:00   | 02:50:00    | 00:10:00
  67890 | frank   |         1003 | mba           | 2021-11-07 | first interview  |  56789 | 2021-11-05 | 02:40:00   | 02:50:00    | 00:10:00
  23456 | marker  |         1004 | btech         | 2021-11-04 | second interview |  56789 | 2021-11-05 | 02:40:00   | 02:50:00    | 00:10:00
  12678 | blaze   |         1005 | mtech         | 2021-11-02 | accepted         |  56789 | 2021-11-05 | 02:40:00   | 02:50:00    | 00:10:00
  34567 | clare   |         1006 | btech         | 2021-11-06 | active           |  56789 | 2021-11-05 | 02:40:00   | 02:50:00    | 00:10:00
  12345 | jonas   |         1001 | btech         | 2021-11-01 | active           |  67890 | 2021-11-07 | 11:30:00   | 11:50:00    | 00:20:00
  56789 | scarlet |         1002 | mtech         | 2021-11-05 | offer            |  67890 | 2021-11-07 | 11:30:00   | 11:50:00    | 00:20:00
  67890 | frank   |         1003 | mba           | 2021-11-07 | first interview  |  67890 | 2021-11-07 | 11:30:00   | 11:50:00    | 00:20:00
  23456 | marker  |         1004 | btech         | 2021-11-04 | second interview |  67890 | 2021-11-07 | 11:30:00   | 11:50:00    | 00:20:00
  12678 | blaze   |         1005 | mtech         | 2021-11-02 | accepted         |  67890 | 2021-11-07 | 11:30:00   | 11:50:00    | 00:20:00
  34567 | clare   |         1006 | btech         | 2021-11-06 | active           |  67890 | 2021-11-07 | 11:30:00   | 11:50:00    | 00:20:00
  12345 | jonas   |         1001 | btech         | 2021-11-01 | active           |  23456 | 2021-11-04 | 12:30:00   | 01:30:00    | 01:00:00
  56789 | scarlet |         1002 | mtech         | 2021-11-05 | offer            |  23456 | 2021-11-04 | 12:30:00   | 01:30:00    | 01:00:00
  67890 | frank   |         1003 | mba           | 2021-11-07 | first interview  |  23456 | 2021-11-04 | 12:30:00   | 01:30:00    | 01:00:00
  23456 | marker  |         1004 | btech         | 2021-11-04 | second interview |  23456 | 2021-11-04 | 12:30:00   | 01:30:00    | 01:00:00
  12678 | blaze   |         1005 | mtech         | 2021-11-02 | accepted         |  23456 | 2021-11-04 | 12:30:00   | 01:30:00    | 01:00:00
  34567 | clare   |         1006 | btech         | 2021-11-06 | active           |  23456 | 2021-11-04 | 12:30:00   | 01:30:00    | 01:00:00
  12345 | jonas   |         1001 | btech         | 2021-11-01 | active           |  12678 | 2021-11-02 | 03:40:00   | 04:10:00    | 00:30:00
  56789 | scarlet |         1002 | mtech         | 2021-11-05 | offer            |  12678 | 2021-11-02 | 03:40:00   | 04:10:00    | 00:30:00
  67890 | frank   |         1003 | mba           | 2021-11-07 | first interview  |  12678 | 2021-11-02 | 03:40:00   | 04:10:00    | 00:30:00
  23456 | marker  |         1004 | btech         | 2021-11-04 | second interview |  12678 | 2021-11-02 | 03:40:00   | 04:10:00    | 00:30:00
  12678 | blaze   |         1005 | mtech         | 2021-11-02 | accepted         |  12678 | 2021-11-02 | 03:40:00   | 04:10:00    | 00:30:00
  34567 | clare   |         1006 | btech         | 2021-11-06 | active           |  12678 | 2021-11-02 | 03:40:00   | 04:10:00    | 00:30:00
  12345 | jonas   |         1001 | btech         | 2021-11-01 | active           |  34567 | 2021-11-06 | 02:00:00   | 03:00:00    | 01:00:00
  56789 | scarlet |         1002 | mtech         | 2021-11-05 | offer            |  34567 | 2021-11-06 | 02:00:00   | 03:00:00    | 01:00:00
  67890 | frank   |         1003 | mba           | 2021-11-07 | first interview  |  34567 | 2021-11-06 | 02:00:00   | 03:00:00    | 01:00:00
  23456 | marker  |         1004 | btech         | 2021-11-04 | second interview |  34567 | 2021-11-06 | 02:00:00   | 03:00:00    | 01:00:00
  12678 | blaze   |         1005 | mtech         | 2021-11-02 | accepted         |  34567 | 2021-11-06 | 02:00:00   | 03:00:00    | 01:00:00
  34567 | clare   |         1006 | btech         | 2021-11-06 | active           |  34567 | 2021-11-06 | 02:00:00   | 03:00:00    | 01:00:00
(36 rows)
```

26.   INNER JOIN

Syntax:
SELECT column_name(s)
FROM table1
INNER JOIN table2
ON table1.column_name = table2.column_name;

Query: Combine the data from applicants table and attendance table and retrieve the records from both the tables.

```
hrms=# select applicants.name,applicants.candidate_no,attendance.date FROM applicants INNER JOIN attendance ON applicants.app_id=attendance.att_id ;
  name   | candidate_no |    date
---------+--------------+------------
 jonas   |         1001 | 2021-11-01
 scarlet |         1002 | 2021-11-05
 frank   |         1003 | 2021-11-07
 marker  |         1004 | 2021-11-04
 blaze   |         1005 | 2021-11-02
 clare   |         1006 | 2021-11-06
(6 rows)
```

27.   OUTER JOINING

Syntax :
SELECT column_name(s)
FROM table1
FULL OUTER JOIN table2
ON table1.column_name = table2.column_name
WHERE condition;

Query : Perform a full outer join on tables applicants and attendance which returns the output that matches the id of applicants with the attendance id.

```
hrms=# select applicants.name,attendance.date FROM applicants FULL OUTER JOIN attendance ON applicants.app_id=attendance.att_id ORDER BY applicants.name;
  name   |    date
---------+------------
 blaze   | 2021-11-02
 clare   | 2021-11-06
 frank   | 2021-11-07
 jonas   | 2021-11-01
 marker  | 2021-11-04
 scarlet | 2021-11-05
(6 rows)
```

28.   ALIASING

Syntax :
SELECT column_name AS alias_name
FROM table_name;

Query : Fetch app_status from applicants table using status as alias name

```
hrms=# select app_status AS STATUS FROM applicants;
      status
------------------
 active
 offer
 first interview
 second interview
 accepted
 active
(6 rows)
```

29.   VIEW

Syntax :
CREATE VIEW view_name AS
SELECT column1, column2, ...
FROM table_name
WHERE condition;

Query: Create a view for the applicants whose qualification is btech

```
hrms=# create view detailsview AS select name,qualification from  applicants where qualification='btech';
CREATE VIEW
hrms=# select*from detailsview;
  name  | qualification
--------+---------------
 jonas  | btech
 marker | btech
 clare  | btech
(3 rows)
```

30. CREATE INDEX

Syntax :

CREATE INDEX index_name

ON table_name (column1, column2, ...);

Query : Create an index with attributes applicants number on a table vacancies.

```
hrms=# create index applicantsnumber on vacancies(dept_id);
CREATE INDEX
hrms=# \ci+ applicantsnumber;
invalid command \ci+
Try \? for help.
hrms=# \di+ applicantsnumber;
                              List of relations
 Schema |        Name        | Type  | Owner   |   Table   | Persistence | Size  | Description
--------+--------------------+-------+---------+-----------+-------------+-------+-------------
 public | applicantsnumber   | index | postgres | vacancies | permanent  | 16 kB |
(1 row)


hrms=# drop index applicantsnumber;
DROP INDEX
hrms=# \di+ applicantsnumber;
Did not find any relation named "applicantsnumber".
```

31. DROP INDEX

Syntax:

 DROP INDEX index_name;

 Query: Drop all the indexes created on table vacancies.

```
hrms=# create index applicantsnumber on vacancies(dept_id);
CREATE INDEX
hrms=# \ci+ applicantsnumber;
invalid command \ci+
Try \? for help.
hrms=# \di+ applicantsnumber;
                              List of relations
 Schema |        Name        | Type  | Owner   |   Table   | Persistence | Size  | Description
--------+--------------------+-------+---------+-----------+-------------+-------+-------------
 public | applicantsnumber   | index | postgres | vacancies | permanent  | 16 kB |
(1 row)


hrms=# drop index applicantsnumber;
DROP INDEX
hrms=# \di+ applicantsnumber;
Did not find any relation named "applicantsnumber".
```
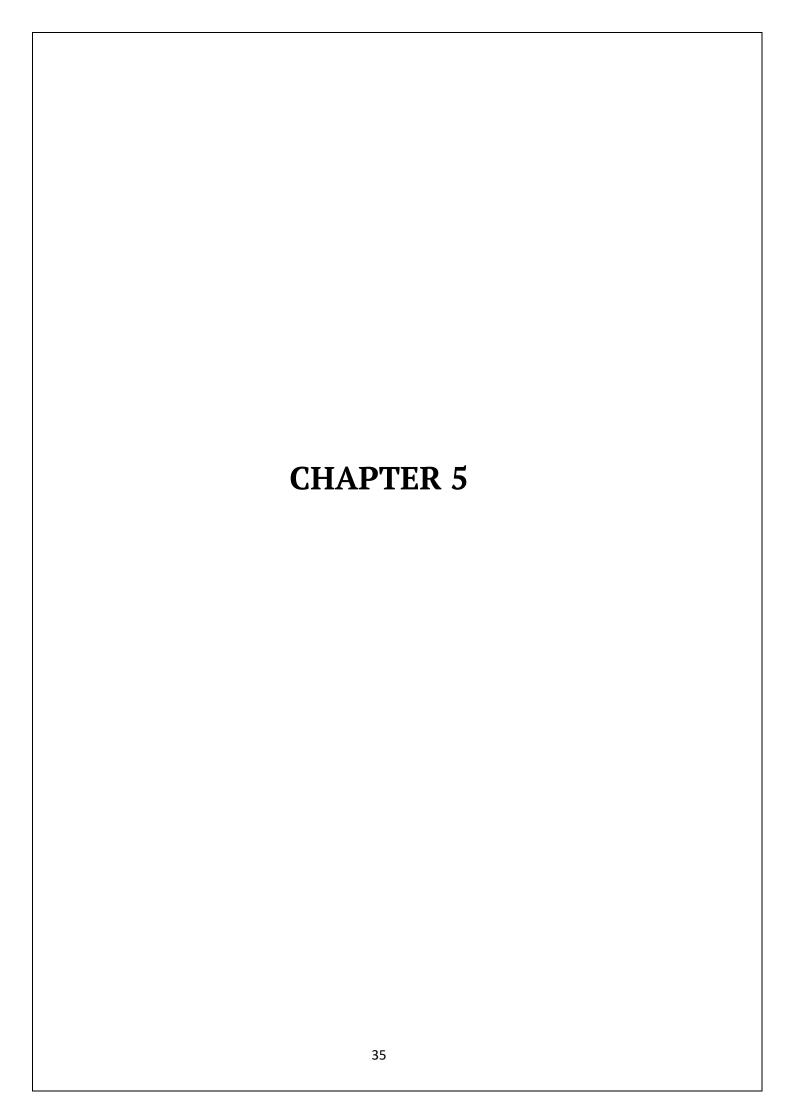
CHAPTER 5

CONCLUSION:

Hence, we have successfully implemented a human resource management and payroll system.

HRMS-Human Resource Management System is undoubtedly a great shield to make your way through the fierce competition in the market. It allows the employees and the admin to explore more possibilities in a user-friendly manner.

From expenses to attendance, timesheets, audits and projects, a good HRMS software will assist an Engineering Company to utilise its maximum potential and enhance the functioning and productivity of its enterprise.

With the constant changes in the work structure of the corporate world, remote work is evolving as a trend that is here to stay. A promising HRMS system is the medium to adapt to this trend and get its assistance to scale the heights of success and make a name in the industry.

We have successfully implemented various functionalities and created the fully functional database management system for human resources.

SOFTWARE USED:

PostgreSQL