# VISVESVARAYA TECHNOLOGICAL UNIVERSITY

**“Jnana Sangama”, Belgaum-590018.**

**DBMS MINI PROJECTREPORT**

ON

# Result Analysis Management System

***Submitted in Partial fulfilment for the V Semester, BE, Information Science & Engineering.***

**Submitted by:**

| | |
|---|---|
| **B.U.KAVYA** | **1GA17IS005** |

**Under the Guidance of:**

**Ms. Sharmila Chidaravalli**<br>
**Assistant Professor**<br>
**Dept. of IS&E**

**2019-2020**

**DEPARTMENT OFINFORMATION SCIENCE&ENGINEERING**

## GLOBAL ACADEMY OF TECHNOLOGY

**Ideal Homes, RR Nagar, Bengaluru-560098**

**(Affiliated to Visvesvaraya Technological University, Belgaum and Approved by AICTE, New Delhi, NAAC Accredited with ‘A’ Grade)**

# GLOBAL ACADEMY OF TECHNOLOGY

**Ideal Homes, RR Nagar, Bengaluru-560098.**

**(Affiliated to Visvesvaraya Technological University, Belgaum and Approved by AICTE, New Delhi,**
**NAAC Accredited with , 'A' Grade)**

## Department of Information Science and Engineering

# CERTIFICATE

Certified that the project work entitled "**Result Analysis Management System**" carried out by **B.U.Kavya,** bonafide students of GLOBAL ACADEMY OF TECHNOLOGY, in partial fulfillment for the award of Bachelor of Engineering in Information Science and Engineering of the Visvesvaraya Technological University, Belgaum during the year 2019-20. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report deposited in the department library. The Mini project report has been approved as it satisfies the academic requirements in respect of DBMS with Mini Project(17CSL58) prescribed for the said Degree.

Signature of the Guide
**Sharmila Chidaravalli**
Asst. Prof. Dept. of ISE
GAT, Bengaluru.

Signature of the HOD
**Dr. Ganga Holi**
Prof. & Head, Dept. of ISE
GAT, Bengaluru.

**External Viva**

**Name of the Examiners** **Signature with date**

**1.** ----------------------------- -----------------------------

**2.** ----------------------------- -----------------------------

# ABSTRACT

Result analysis management system is a web based application developed for college to analyze the result and keep track of student record. We can view the individual candidate's result separately. This system has been designed to carry out marks analysis process in the educational institution. It comprises of university seat number, student name, subject wise both internal and external marks, grand total, percentage and result. This data can be accessed by both student and faculties. Students have only read access on the data. Faculties have both read and write access on the data, they can update, delete or display the records. It ensures secured data access by providing the login and username for teachers portal. This project also maintains faculty details like their name, email-id, date of birth and other fields. We have used triggers to maintain a class teacher table and stored procedures to calculate the age of faculties automatically.

# ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crowned our effort with success.

We are grateful to our institution, **Global Academy of Technology**, with its ideals and inspirations for having provided us with the facilities, which has made this project a success.

We earnestly thank **Dr. Rana Pratap Reddy,Principal ,Global Academy of Technology** for facilitating academic excellence in the college and providing us with the congenial environment to work in, that helped us in completing this project.

We wish to extend our profound thanks to **Dr.Ganga Holi, Prof.& Head, Department of Information Science & Engineering, GAT,** for giving us the consent to carry out this project.

We would like to express our sincere thanks to our internal guide **Ms. Sharmila Chidaravalli, Assistant professor, Department of Information Science & Engineering, GAT,** for her able guidance and valuable advice at every stage, which helped us in the successful completion of the project.

We owe our sincere thanks to **Mrs. Lakshmi R, Assistant professor, Department of Information Science & Engineering, GAT ,** for her immense help in carrying out the project and for her valuable suggestions on the project.

We owe our sincere thanks to **Mr. Sridhar R, Assistant professor, Department of Information Science & Engineering, GAT ,** for his guidance.

We would like to thank all the teaching and non-teaching staff for their valuable advice and support.

We would like to express our sincere thanks to our parents and friends for their support.

**B.U.Kavya**

# CONTENTS

# LIST OF FIGURES

**Figure No.** **Page No.**

# CHAPTER 1

# INTRODUCTION

The main objective of the project is to provide the examination results in a simpler way. This project is useful for both students and faculties for getting the results in a more efficient way .privileges provided for a student is to read his/her result by providing their university seat number(USN) and privileges provided for a faculty is delete, update search for a student’s record by providing the USN or the name of a student.

Additionally teachers can view the top 10 scores of a class, categorize the students based on their results (distinction, first class , second class, failures).faculties are provided with their username and password to login. So, this ensures more securing way of accessing the data.

We have used MYSQL to store the database records at the back end and Eclipse IDE along with jsp codes written in it as the front end to get the databases connectivity.

Apache Tomcat is the used to connect between the web pages and the jsp pages. It provides a "pure Java" HTTP web server environment in which Java code can run.

On using the above mentioned software, a user friendly webpage is created so that the user can interact with the database on basis of his desire.

# CHAPTER 2

# LITERATURE SURVEY

## EXISTING SYSYTEM/ALGORITHM

The Student Database System is the software that enables user to easily store and find record's information such as name, roll no and marks. There are record-centric databases (database.txt and record.txt) that provide a fully integrated approach to store information from user and communicate with the software.

## DISADVANTAGE OF EXISTING SYSTEM/ALGORITHM

The main disadvantage of this software is that it is OS dependent since it is made in C programming language. It is not portable like other languages like JAVA whose software can run on any operating system. Also, it is console based software so we cannot use attractive features which are used in window, web or mobile application. The use of linear search in file handling might increase the time complexity.

## PROPOSED APPROACH AND ITS JUSTIFICATION

The Student Database System may be chosen because it is thought to provide following advantages: -

1. This software is space and time efficient.
2. It is small and user friendly.
3. Document Management.
4. File Handling is effectively implemented.
5. Attractive design.

# CHAPTER 3

# SYSTEM REQUIREMENTS AND SPECIFICATION

Every project has its own specifications with respect to the requirements and configurations. This may be in the form of hardware and software requirements or functional and non-functional requirements.

## 2.1 Hardware Requirements

- Processor Intel® core™ i7 CPU Q720 1.60GHz
- Operating System : windows 7 Home Premium
- RAM : 4 GB
- Type : 64-bit Operating System
- CPU Clock : 1.60GHz

## 3.1 Software Requirements

- Operating system- Windows 10 is used as the operating system as it is stable and supports more features and is more user friendly.
- Database MYSQL-MYSQL is used as database as it easy to maintain and retrieve records by simple queries which are in English language which are easy to understand and easy to write.

- **SQL**

  SQL is a domain-specific language used in programming and designed for managing data held in a relational database management system, or for stream processing in a relational data stream management system.

- **Eclipse IDE**

  Eclipse is an integrated development environment (IDE) used in computer programming, and is the most widely used Java IDE. It contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in Java and its primary use is for developing Java applications.

## Dependencies for eclipse ide:

**JDK:** The Java Development Kit (JDK) is an implementation of either one of the Java Platform, Standard Edition, Java Platform, Enterprise Edition, or Java Platform, Micro Edition platforms[1] released by Oracle Corporation in the form of a binary product aimed at Java developers on Solaris, Linux, macOS or Windows.

## MYSQL

MYSQL Database was used for populating all the tables. MSQL is a relational database management System (RDBMS), and ships with no GUI tool to administer MYSQL database or manage data contained within the databases. Users may use the included command line tools or use MYSQL “front ends”, desktop software and web applications that create and manage. MYSQL database, build database structures, back up data, inspect status, and work with data records. The official set of MYSQL front-end tools, MYSQL Workbench is actively developed by XAMPP and is freely available for use. MYSQL ships with many command line tools, from which the main interface is ‘MySQL’ client Third parties have also developed tools to manage MYSQL servers.

## Apache Tomcat

It is an open source implementation of the Java Servlet, Java Server Pages, Java Expression Language and Web Socket technologies. Tomcat provides a "pure Java" HTTP web server environment in which Java code can run.

Tomcat is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation, released under the Apache License 2.0 license.

## HTML

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

## JSP

Java Server Pages (JSP) is a collection of technologies that helps software developers create dynamically generated web pages based on HTML, XML, SOAP, or other document types. Released in 1999 by Sun Microsystems, JSP is similar to PHP and ASP, but it uses the Java programming language**.**

# CHAPTER 4

# SYSTEM DESIGN

Introduction to Database Design Model Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer’s goal is to produce a model or representation of an entity that will later be built. Beginning, once system requirement have been specified and analysed, system design is the first of the three technical activities design, code and test that is required to build and verify software. The importance can be stated with a single word “Quality”. Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer’s view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system, one that will be difficult to test, and one whose quality cannot be assessed until the last stage. During design, progressive refinement of data structure, program structure, and procedural details are developed reviewed and documented. System design can be viewed from either technical or project management perspective. From the technical point of view, design is comprised of four activities architectural design, data structure design, interface design and procedural design.

# 4.1 ER DIAGRAM

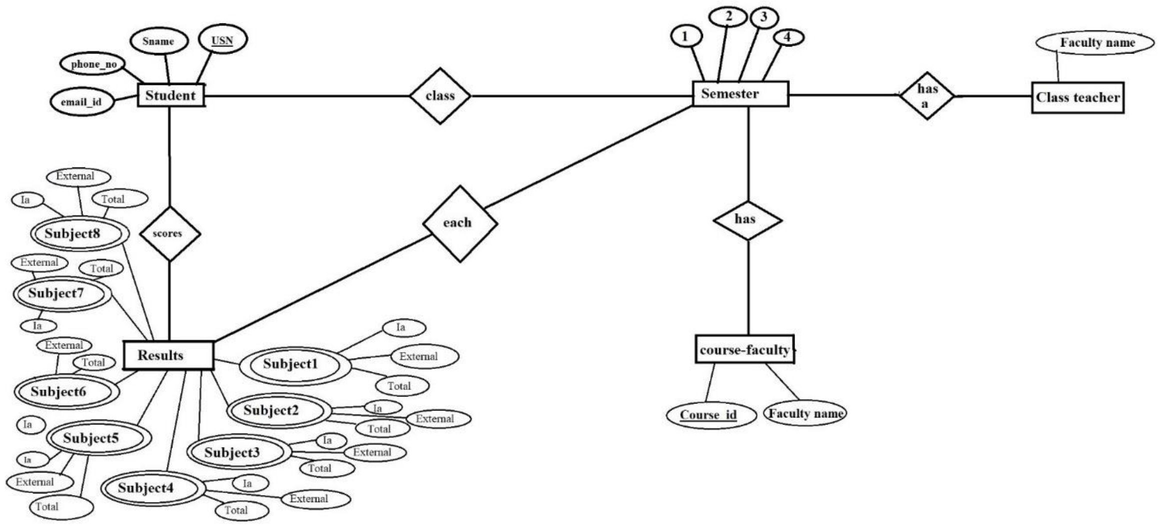


Fig 4.1: Entity Relation Diagram

## 4.2 RELATIONAL SCHEMA DIAGRAM

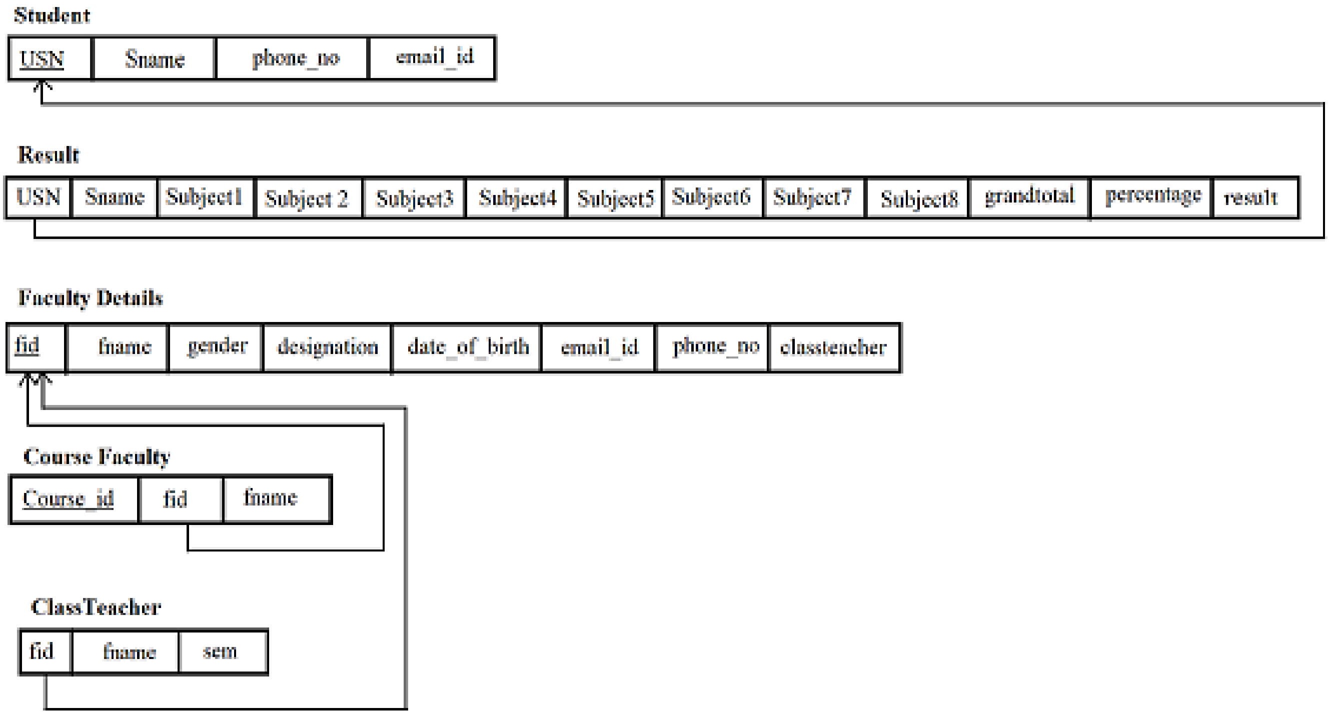


Fig 4.2: Relational Schema Diagram

## 4.3 TEACHERS'S PORTAL

```
<html>

<img src='C:\Users\ADMIN\Desktop\image.jpeg' style='position:fixed;z-index:-
1;width:100%; height:100%;'>

<title>Teacher portal</title>

<h1> TEACHER PORTAL</h1>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<body>

<form action="optiondisplay.jsp">

<h3>SELECT A SEMESTER</h3>

<input type="radio" name="SEM" value="1">1<br>

<input type="radio" name="SEM" value="2">2<br>

<input type="radio" name="SEM" value="3">3<br>

<input type="radio" name="SEM" value="4">4<br>

<input type="radio" name="SEM" value="5">5<br>

<input type="radio" name="SEM" value="6">6<br>

<input type="radio" name="SEM" value="7">7<br>

<input type="radio" name="SEM" value="8">8<br>

<input  type="submit" value="Submit"/>

<input type="reset" value="Reset"/>

</form>

</body>

</html>
```

## 4.4 CREATION AND LOAD

create code for the table 2017_sem4→

```
create table 2017_sem4 (usn varchar(10) primary key,sname char(50),
    17mat41_ia int,17mat41_ext int, 17mat41_tot int,
    17cs42_ia int,17cs42_ext int,17cs42_tot int,
    17cs43_ia int,17cs43_ext int,17cs43_tot int,
     17cs44_ia int,17cs44_ext int,17cs44_tot int,
     17cs45_ia int,17cs45_ext int,17cs45_tot int,
     17cs46_ia int,17cs46_ext int,17cs46_tot int,
     17csl47_ia int,17csl47_ext int,17csl47_tot int,
     17csl48_ia int,17csl48_ext int,17csl48_tot int,
     17kkm49_ia int,17kkm49_ext int,17kkm49_tot int,
     grandtotal int,percentage float,result char(25));
```

The fields mentioned in the table creation must correspond to the number of columns in the excel sheet. We have used 'LOAD' command to insert data into the table. LOAD command loads the data from the excel sheet to the table in MySQL.

LOAD command:

```
load data local infile "E:/our class/4thsem.csv" into table globalise.2017_sem4
fields terminated by ','
lines terminated by '\n';
```

The LOAD DATA INFILE statement reads rows from a text file into a table at a very high speed. If the LOCAL keyword is specified, the file is read from the client host. If LOCAL is not specified, the file must be located on the server. ( LOCAL is available in MySQL 3.22.

## 4.5 CREATION OF TRIGGERS

A trigger is procedural code that is automatically executed in response to certain events on a particular table or view in a database. The trigger is mostly used for maintaining the integrity of the information on the database. If the entered details of the faculty who is the class teacher then the trigger automatically inserts the record into a table where details of class teacher is maintained.

```
delimiter $$
create trigger classteach after insert on faculty_details
for each row
begin
if(new.classteacher) then
insert into classteacher(f_id,fname,sem) values(new.fid,new.fac_name,new.classteacher);
end if;
end $$
```

## 4.6 CREATION OF STORED PROCEDURES

A stored procedure is a subroutine available to applications that access a relational database management system (RDBMS). Such procedures are stored in the database data dictionary. Uses for stored procedures include data-validation or access control mechanisms. Furthermore, stored procedures can consolidate and centralize logic that was originally implemented in applications. Stored procedures can access or modify data in a database, but it is not tied to a specific database or object, which offers a number of advantages. Stored procedure is used to calculate the age of a faculty when the date of birth is provided.

```
DELIMITER $$
CREATE PROCEDURE Getage( )
BEGIN
SELECT * , year(CURRENT_DATE( )) – year(date_of_birth) as AGE FROM
faculty_details;
END$$
```

# CHAPTER 5

# IMPLEMENTATION

Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the users, which it will work efficiently and effectively.

## 5.1 To Connect to MySQL Using JDBC:

The fundamental steps involved in the process of connecting to a database consist of the following:

- Import JDBC packages.
- Register the JDBC driver.
- Open a connection to the database.
- Create statement and execute queries
- Close the connection.

### Import JDBC Packages

The following import statement should be included in the program irrespective of the JDBC driver being used:

➔ import java.sql.*;

### Register the JDBC driver

The forName( ) method of Class is used to register the driver class. This method is used to dynamically load the driver class.

➔ Class.forName(driverName);

➔ String driverName = "com.mysql.jdbc.Driver";

### Open a connection to the database

Once the required packages have been imported and the Oracle JDBC driver has been loaded and registered, a database connection must be established. This is done by using the getConnection() method of the DriverManager class. A call to this method creates an object instance of the java.sql.Connection class.

The getConnection() requires three input parameters, namely, a connect string, a username, and a password. The connect string should specify the JDBC driver to be yes and the database instance to connect to. The following lines of code illustrate using the getConnection() method.

- ➔ Connection conn = DriverManager.getConnection(URL, username, passwd);
- ➔ Connection conn = DriverManager.getConnection(URL);

### Create statement and execute queries

The next step after the JDBC is loaded and connection is successfully made with a particular database managed by the dbms, is to end a particular query to the DBMS for processing.

• SQL query consists series of SQL command that direct DBMS to do something example Return rows.

• Connect.createStatement() method is used to create a statement Object.

• The statement object is then used to execute a query and return result object that contain response from the DBMS

- ➔ statement.executeUpdate(query);
- ➔ resultSet= stmt.executeQuery();

### Closing the Connection

The last step is to close the database connection opened in the beginning after importing the packages and loading the JDBC drivers. This is done by a call to the close() method of the Connection class. The following line of code does this:

- ➔ conn.close();

## 5.2 FRONT END CODES

### 5.2.1 JSP Code for login controller

```
<%@ page import ="java.sql.*" %>
<%@page import="java.sql.Statement"%>
<%@page import="java.sql.Connection"%>
<%
        String un = request.getParameter("username");
        String pw = request.getParameter("password");
         session.setAttribute("username",un );
        try {
            Class.forName("com.mysql.jdbc.Driver");
// loads driver
            Connection c =
DriverManager.getConnection("jdbc:mysql://localhost:3306/globalise","root",
"root"); // gets a new connection
            PreparedStatement ps = c.prepareStatement("select uname,pwd from
members where uname=? and pwd=?");
            ps.setString(1, un);
            ps.setString(2, pw);
            ResultSet rs = ps.executeQuery();
            while (rs.next()) {
                response.sendRedirect("Teacherportal.jsp");
                return;
            }
            response.sendRedirect("LOGIN.jsp");
            return;
        } catch (ClassNotFoundException | SQLException e) {
            out.println("Wrong username and password! <br>Don't have an
account?<br>");
        }%>
```

### 5.2.2 JSP Code for registration controller

```
<%@ page import ="java.sql.*" %>
<%
try{    String user = request.getParameter("uname");
    String pwd = request.getParameter("pass");
   out.println(user+pwd);
    Class.forName("com.mysql.jdbc.Driver");
    Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/globalise","root", "root");
    Statement st = con.createStatement();
    //ResultSet rs;
    int i = st.executeUpdate("insert into members(uname, pwd) values ('"+ user + "','" +
pwd + "')");
    if (i > 0) {
       // session.setAttribute("userid", user);
        response.sendRedirect("LOGIN.jsp");
       // out.print("Registration Successfull!"+"<a href='index.jsp'>Go to Login</a>");
    } else {
        response.sendRedirect("error.jsp");
    }
}
catch(Exception e)
{
            e.printStackTrace();
}
%>
```

### 5.2.3 JSP Code for displaying top 10 students

```
<html>
<%@page import="java.sql.DriverManager"%>
<%@page import="java.sql.*"%>
<%@page import="java.sql.Statement"%>
<%@page import="java.sql.Connection"%>
<%!String sem,query,q1;
int i;%>
<%
String id = request.getParameter("userId");
String driverName = "com.mysql.jdbc.Driver";
String connectionUrl = "jdbc:mysql://localhost:3306/";
String dbName = "globalise";
String userId = "root";
String password = "root";
try {
Class.forName(driverName);
} catch (ClassNotFoundException e) {
e.printStackTrace();
}
Connection connection = null;
Statement statement = null;
ResultSet resultSet = null;
%>
<h2>TOP 10 STUDENTS ARE:</h2>
```

```
<table align="center" cellpadding="5" cellspacing="5" border="1">
<tr>
</tr>
<tr  bgcolor="#FFC300">
<%int i=3,j=0;
String sem=(String) session.getAttribute("Semester");
try{
connection = DriverManager.getConnection(connectionUrl+dbName, userId, password);
statement=connection.createStatement();
query="select usn,sname,grandtotal,percentage,result from 2017_sem"+sem+" order by
grandtotal desc limit 10;";
resultSet= statement.executeQuery(query);
ResultSetMetaData rsmd = resultSet.getMetaData();
int count = rsmd.getColumnCount();
for(int v=1;v<=count;v++)
{
        %><td><%=rsmd.getColumnName(v)%>
        <%
}
%></tr><%
while(resultSet.next()){
%>
<tr bgcolor="#DEB887">
<td><%=resultSet.getString("usn") %></td>
<td><%=resultSet.getString("sname") %></td>
<td><%=resultSet.getInt("grandtotal") %></td>
<td><%=resultSet.getFloat("percentage") %></td>

<td><%=resultSet.getString("result") %></td>

</tr>

<%  }

%></table>

<br>

<br>

<br>

<input type="button" value="back" onclick="history.back()">

 <%

} catch (Exception e) {

e.printStackTrace();

}

%>

</html>
```

## 5.2.4 HTML Code for login page

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

   <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

   <title>Login page</title>

</head>

<body  background="C:\Users\ADMIN\Desktop\image.jpeg" height="100%">
```

```
LOG IN PAGE WELCOME

<br>

<br>

<br>

<form action="logincontroller.jsp" method="post">

   Enter username :<input type="text" name="username"> <br>

  Enter password :<input type="password" name="password"><br>

 <input type="submit" value="Login">

</form>

<form action="register.jsp" method ="post">

<input type="submit"  value="Create new account">

<br>

<br>

<br>

<br>

<h5> NOT A TEACHER? CLICK HERE FOR <a href="Studentportal.jsp">STUDENT PORTAL</a></h5>

</form>

</body>

</html>
```

## 5.2.5 HTML Code for register page

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

    <head>
```

```
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Registration</title>
    </head>
    <body>
        <form method="post" action="registrationcontroller.jsp">
            <center>
            <table border="1" width="30%" cellpadding="5">
                <thead>
                    <tr>
                        <th colspan="2">Enter Information Here</th>
                    </tr>
                </thead>
                <tbody>
                    <tr>
                        <td>User Name</td>
                        <td><input type="text" name="uname" value="" /></td>
                    </tr>
                    <tr>
                        <td>Password</td>
                        <td><input type="password" name="pass" value="" /></td>
                    </tr>
                    <tr>
                        <td><input type="submit" value="Submit" /></td>
                        <td><input type="reset" value="Reset" /></td>
                    </tr>
                    <tr>
                        <td colspan="2">Already registered? <a href="LOGIN.jsp">Login
    Here</a></td>
                    </tr>
                </tbody>
            </table>
            </center>
        </form>
    </body>
    </html>
```

## 5.2.6 HTML Code for search page

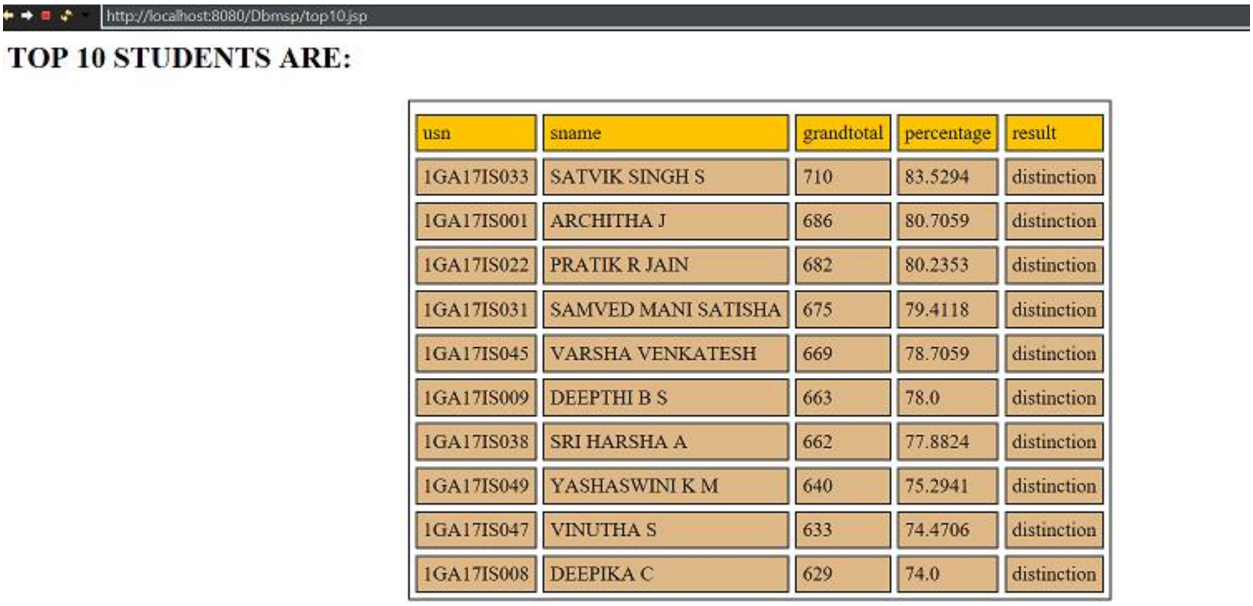
```
<html>
<h3>SEARCH FOR:</h3>
<body>
<form method="get" action="particularstu.jsp">
<h3>search for a particular student?</h3>
<br>
<h3>enter USN:</h3><input type="text" name="usn"/>
  <button type="submit">SEARCH STUDENT</button>
</form>
<br>
<form method="get" action="top10.jsp">

  <button type="submit">TOP 10</button>
</form>
```

```
<br>

<br>

<br>

<form method="get" action="passpercentage.jsp">


    <button type="submit">PASS PERCENTAGE</button>

</form>

<br>

<br>

<br>

<form method="get" action="distinction.jsp">


    <button type="submit">DISTINCTION</button>

</form>

<br>

<br>

<br>

<form method="get" action="firstclass.jsp">


    <button type="submit">FIRST CLASS</button>

</form>

<br>

<br>

<br>

<form method="get" action="secondclass.jsp">
```

```
    <button type="submit">SECOND CLASS</button>

</form>

<br>

<br>

<br>

<form method="get" action="fail.jsp">

    <button type="submit">FAIL</button>

</form>

</html>
```

# CHAPTER 6

# RESULTS

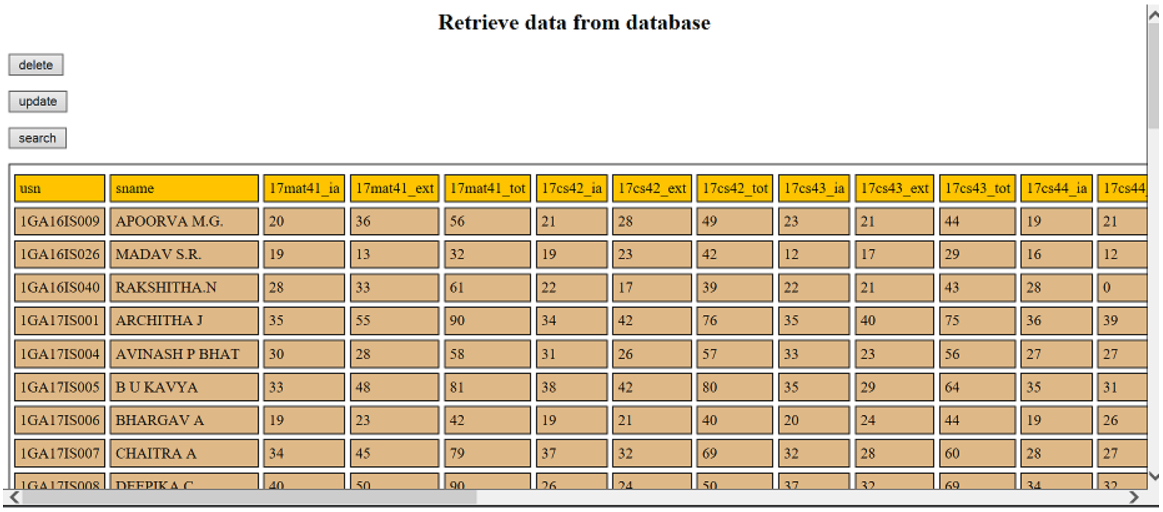


http://localhost:8080/Dbmsp/top10.jsp

TOP 10 STUDENTS ARE:

| usn | sname | grandtotal | percentage | result |
|---|---|---|---|---|
| 1GA17IS033 | SATVIK SINGH S | 710 | 83.5294 | distinction |
| 1GA17IS001 | ARCHITHA J | 686 | 80.7059 | distinction |
| 1GA17IS022 | PRATIK R JAIN | 682 | 80.2353 | distinction |
| 1GA17IS031 | SAMVED MANI SATISHA | 675 | 79.4118 | distinction |
| 1GA17IS045 | VARSHA VENKATESH | 669 | 78.7059 | distinction |
| 1GA17IS009 | DEEPTHI B S | 663 | 78.0 | distinction |
| 1GA17IS038 | SRI HARSHA A | 662 | 77.8824 | distinction |
| 1GA17IS049 | YASHASWINI K M | 640 | 75.2941 | distinction |
| 1GA17IS047 | VINUTHA S | 633 | 74.4706 | distinction |
| 1GA17IS008 | DEEPIKA C | 629 | 74.0 | distinction |

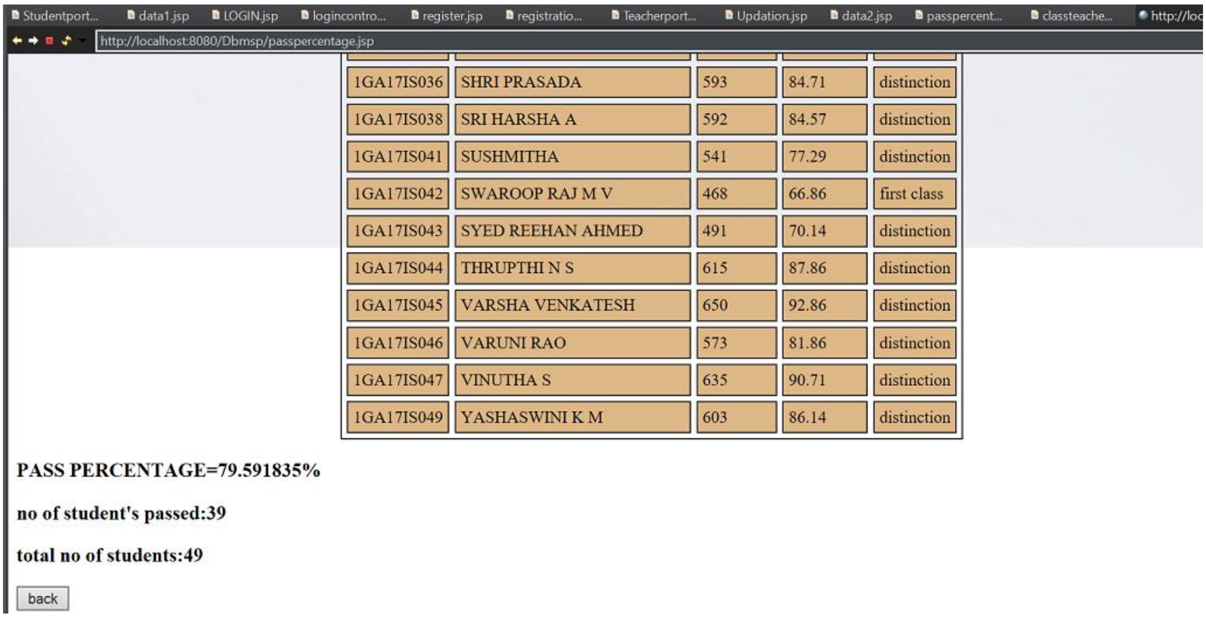
Fig 6.1: Top 10 students retrieved



Retrieve data from database

| usn | sname | 17mat41_ia | 17mat41_ext | 17mat41_tot | 17cs42_ia | 17cs42_ext | 17cs42_tot | 17cs43_ia | 17cs43_ext | 17cs43_tot | 17cs44_ia | 17cs44 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1GA16IS009 | APOORVA M.G. | 20 | 36 | 56 | 21 | 28 | 49 | 23 | 21 | 44 | 19 | 21 |
| 1GA16IS026 | MADAV S.R. | 19 | 13 | 32 | 19 | 23 | 42 | 12 | 17 | 29 | 16 | 12 |
| 1GA16IS040 | RAKSHITHA.N | 28 | 33 | 61 | 22 | 17 | 39 | 22 | 21 | 43 | 28 | 0 |
| 1GA17IS001 | ARCHITHA J | 35 | 55 | 90 | 34 | 42 | 76 | 35 | 40 | 75 | 36 | 39 |
| 1GA17IS004 | AVINASH P BHAT | 30 | 28 | 58 | 31 | 26 | 57 | 33 | 23 | 56 | 27 | 27 |
| 1GA17IS005 | B U KAVYA | 33 | 48 | 81 | 38 | 42 | 80 | 35 | 29 | 64 | 35 | 31 |
| 1GA17IS006 | BHARGAV A | 19 | 23 | 42 | 19 | 21 | 40 | 20 | 24 | 44 | 19 | 26 |
| 1GA17IS007 | CHAITRA A | 34 | 45 | 79 | 37 | 32 | 69 | 32 | 28 | 60 | 28 | 27 |
| 1GA17IS008 | DEEPIKA C | 40 | 50 | 90 | 26 | 24 | 50 | 37 | 32 | 69 | 34 | 32 |

Fig 6.2: Retrieve data from database



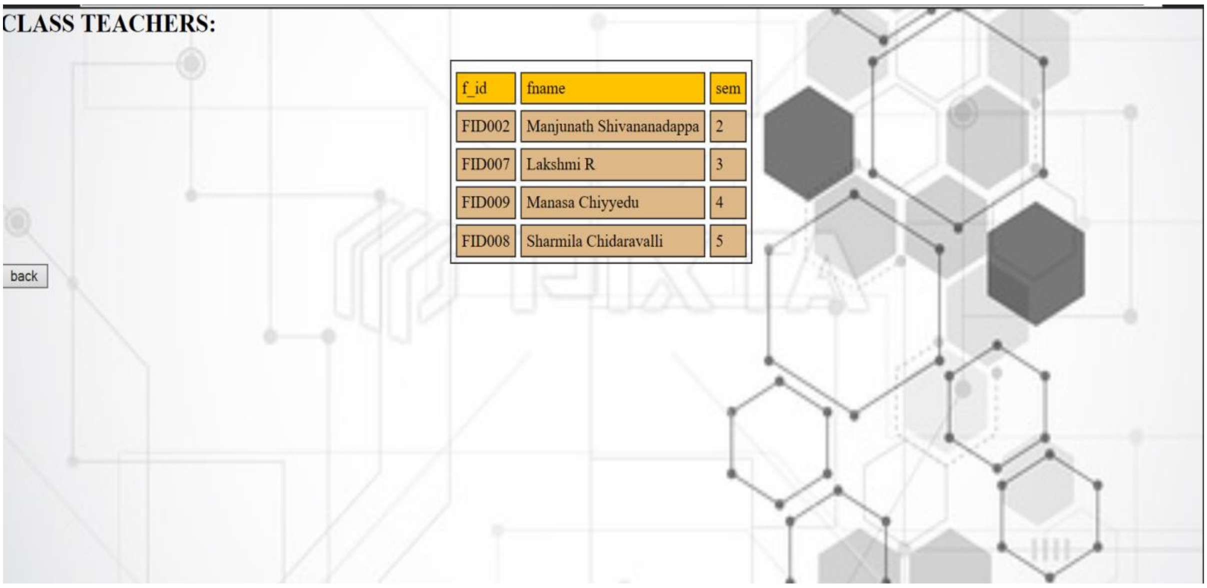
| | | | | |
|---|---|---|---|---|
| 1GA17IS036 | SHRI PRASADA | 593 | 84.71 | distinction |
| 1GA17IS038 | SRI HARSHA A | 592 | 84.57 | distinction |
| 1GA17IS041 | SUSHMITHA | 541 | 77.29 | distinction |
| 1GA17IS042 | SWAROOP RAJ M V | 468 | 66.86 | first class |
| 1GA17IS043 | SYED REEHAN AHMED | 491 | 70.14 | distinction |
| 1GA17IS044 | THRUPTHI N S | 615 | 87.86 | distinction |
| 1GA17IS045 | VARSHA VENKATESH | 650 | 92.86 | distinction |
| 1GA17IS046 | VARUNI RAO | 573 | 81.86 | distinction |
| 1GA17IS047 | VINUTHA S | 635 | 90.71 | distinction |
| 1GA17IS049 | YASHASWINI K M | 603 | 86.14 | distinction |

**PASS PERCENTAGE=79.591835%**

**no of student's passed:39**

**total no of students:49**

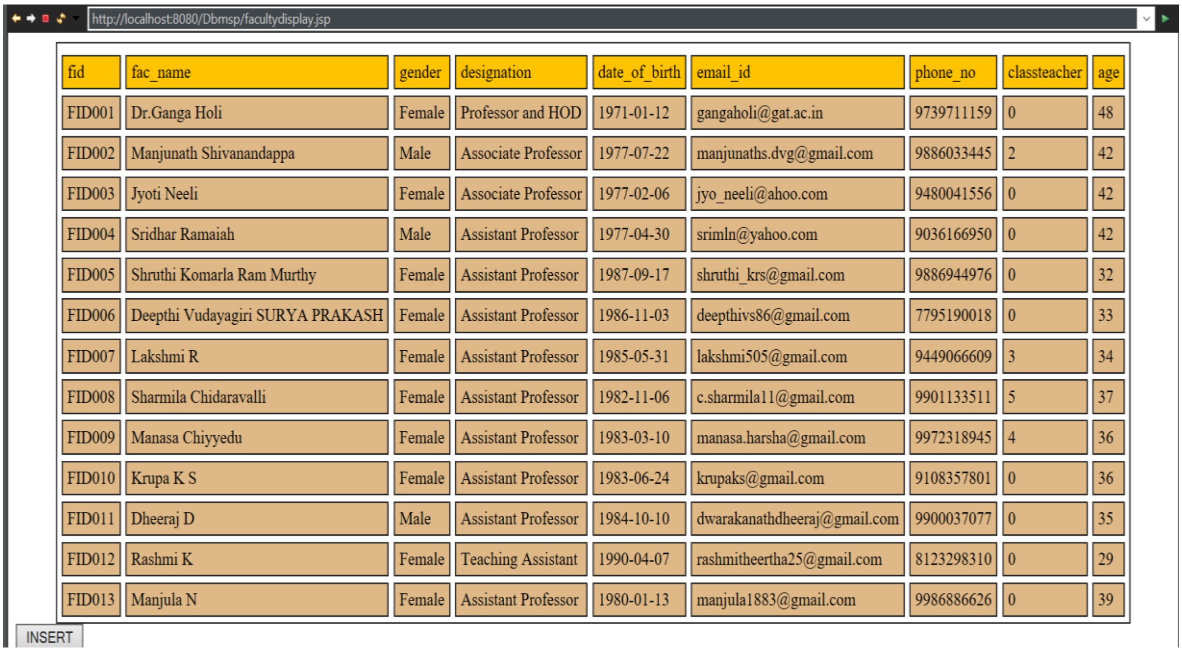
Fig 6.3: Pass percentage calculation



CLASS TEACHERS:

| f_id | fname | sem |
|---|---|---|
| FID002 | Manjunath Shivananadappa | 2 |
| FID007 | Lakshmi R | 3 |
| FID009 | Manasa Chiyyedu | 4 |
| FID008 | Sharmila Chidaravalli | 5 |

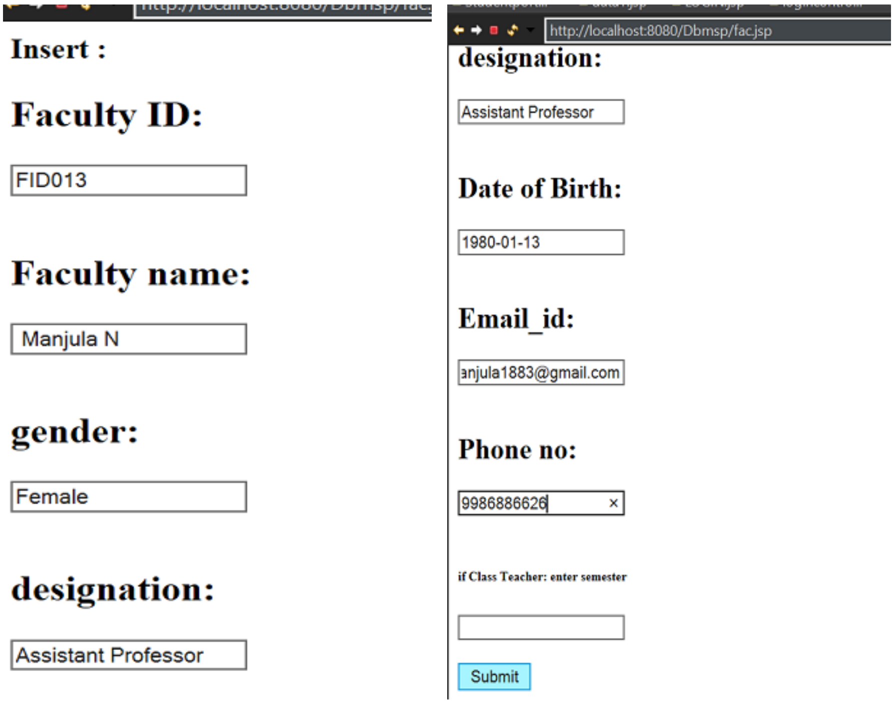
Fig 6.4: Triggers demonstration



http://localhost:8080/Dbmsp/facultydisplay.jsp

| fid | fac_name | gender | designation | date_of_birth | email_id | phone_no | classteacher | age |
|---|---|---|---|---|---|---|---|---|
| FID001 | Dr.Ganga Holi | Female | Professor and HOD | 1971-01-12 | gangaholi@gat.ac.in | 9739711159 | 0 | 48 |
| FID002 | Manjunath Shivanandappa | Male | Associate Professor | 1977-07-22 | manjunaths.dvg@gmail.com | 9886033445 | 2 | 42 |
| FID003 | Jyoti Neeli | Female | Associate Professor | 1977-02-06 | jyo_neeli@ahoo.com | 9480041556 | 0 | 42 |
| FID004 | Sridhar Ramaiah | Male | Assistant Professor | 1977-04-30 | srimln@yahoo.com | 9036166950 | 0 | 42 |
| FID005 | Shruthi Komarla Ram Murthy | Female | Assistant Professor | 1987-09-17 | shruthi_krs@gmail.com | 9886944976 | 0 | 32 |
| FID006 | Deepthi Vudayagiri SURYA PRAKASH | Female | Assistant Professor | 1986-11-03 | deepthivs86@gmail.com | 7795190018 | 0 | 33 |
| FID007 | Lakshmi R | Female | Assistant Professor | 1985-05-31 | lakshmi505@gmail.com | 9449066609 | 3 | 34 |
| FID008 | Sharmila Chidaravalli | Female | Assistant Professor | 1982-11-06 | c.sharmila11@gmail.com | 9901133511 | 5 | 37 |
| FID009 | Manasa Chiyyedu | Female | Assistant Professor | 1983-03-10 | manasa.harsha@gmail.com | 9972318945 | 4 | 36 |
| FID010 | Krupa K S | Female | Assistant Professor | 1983-06-24 | krupaks@gmail.com | 9108357801 | 0 | 36 |
| FID011 | Dheeraj D | Male | Assistant Professor | 1984-10-10 | dwarakanathdheeraj@gmail.com | 9900037077 | 0 | 35 |
| FID012 | Rashmi K | Female | Teaching Assistant | 1990-04-07 | rashmitheertha25@gmail.com | 8123298310 | 0 | 29 |
| FID013 | Manjula N | Female | Assistant Professor | 1980-01-13 | manjula1883@gmail.com | 9986886626 | 0 | 39 |

Fig 6.5: Faculty details and stored procedure



Insert :

Faculty ID:

FID013

Faculty name:

Manjula N

gender:

Female

designation:

Assistant Professor

http://localhost:8080/Dbmsp/fac.jsp

designation:

Assistant Professor

Date of Birth:

1980-01-13

Email_id:

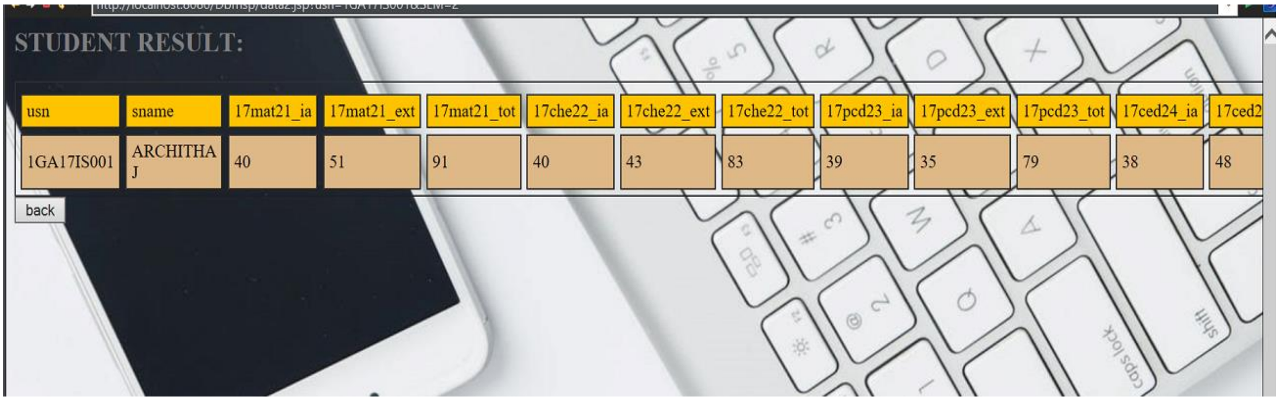
anjula1883@gmail.com

Phone no:

9986886626

if Class Teacher: enter semester

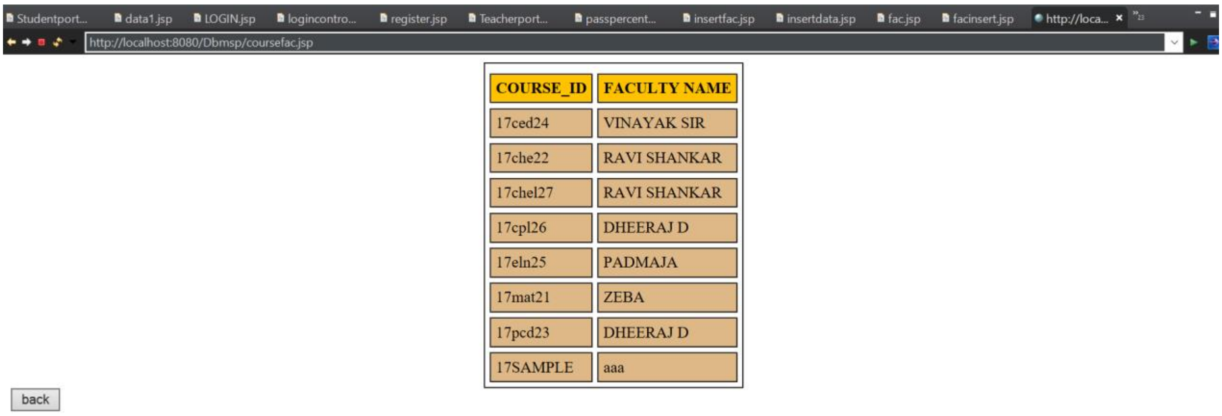
Fig 6.6: Insert demonstration on Faculty table



STUDENT RESULT:

| usn | sname | 17mat21_ia | 17mat21_ext | 17mat21_tot | 17che22_ia | 17che22_ext | 17che22_tot | 17pcd23_ia | 17pcd23_ext | 17pcd23_tot | 17ced24_ia | 17ced2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1GA17IS001 | ARCHITHA J | 40 | 51 | 91 | 40 | 43 | 83 | 39 | 35 | 79 | 38 | 48 |

Fig 6.7: Student portal



http://localhost:8080/Dbmsp/coursefac.jsp

| COURSE_ID | FACULTY NAME |
|---|---|
| 17ced24 | VINAYAK SIR |
| 17che22 | RAVI SHANKAR |
| 17chel27 | RAVI SHANKAR |
| 17cpl26 | DHEERAJ D |
| 17eln25 | PADMAJA |
| 17mat21 | ZEBA |
| 17pcd23 | DHEERAJ D |
| 17SAMPLE | aaa |

Fig 6.8: Relation between course_id and faculty name

# CHAPTER 7

# CONCLUSION

This project is useful for students and institutions for getting the results in simple manner. It minimizes the tedious process of keeping track of Student result data. It provides a much easier access to retrieve the data and it consists of basic operations that are performed on the results such as to retrieve top 10 students, retrieve failed students etc. The project provides Security as the members registered can login which provides read-write access to authorised members and unauthorised members have read only access.

The project further provides details about the faculties handling the courses, Faculties in the department, Class teachers to particular section, the access to these details is provided only to the authorised user. The project is built such that it is user friendly. Analysis of the result system shows details of students by the grade wise result depending on its marks.

# CHAPTER 8

# FUTURE ENHANCEMENT

More functionalities can be added on basis of the user’s requirements. The project can be extended by adding Academic notes and study material link in the student portal, Attendance system in the authorised access, etc. It can be provided with huge amount of data of previous year students and upcoming students to provide an easier access to the data. Project can be provided with a statistical graph that calculates the performance of the student based on the results that he/she has scored. It can be added with facilities such as interaction with the teachers, thus making it more interactive and more user friendly.

# REFERENCES

- https://www.w3schools.com/sql/
- https://en.wikipedia.org/wiki/SQL
- http://www.sqlservertutorial.net
- https://www.tutorialspoint.com
- https://www.geeksforgeeks.org
- https://stackoverflow.com
- https://www.quora.com