INTERNSHIP REPORT

On

Maharashtra Knowledge Corporation Limited (MKCL)

Submitted by

Sayali Prakash Thite

Under the Guidance of
Dr. Rohin Daruwala

DEPARTMENT OF ELECTRONICS ENGINEERING

VEERMATA JIJABAI TECHNOLOGICAL INSTITUTE
(An Autonomous Institute Affiliated to Mumbai University)
(Central Technological Institute, Maharashtra State)
Matunga, MUMBAI - 400019
A.Y. 2022-2023

CERTIFICATE

This is to certify that Sayali Prakash Thite, a student of B. Tech Electronics, has successfully completed the internship at MAHARASHTRA KNOWLEDGE CORPORATION LIMITED (MKCL).

Internship Supervisor	Head of Department
Dr. Rohin Daruwala	Dr. SJ Bhosale
Department Of Electronics,	Department Of Electronics,
VJTI, Mumbai.	VJTI, Mumbai.

Date :

Place : VJTI , Mumbai.

APPROVAL SHEET

The B. Tech internship thesis report for “An Internship Report working at Maharashtra Knowledge Corporation Limited (MKCL)”, submitted by Sayali Prakash Thite - (201061902), is found to be satisfactory and is certified for the graduation of B. Tech Electronics.

Dr. Rohin Daruwala

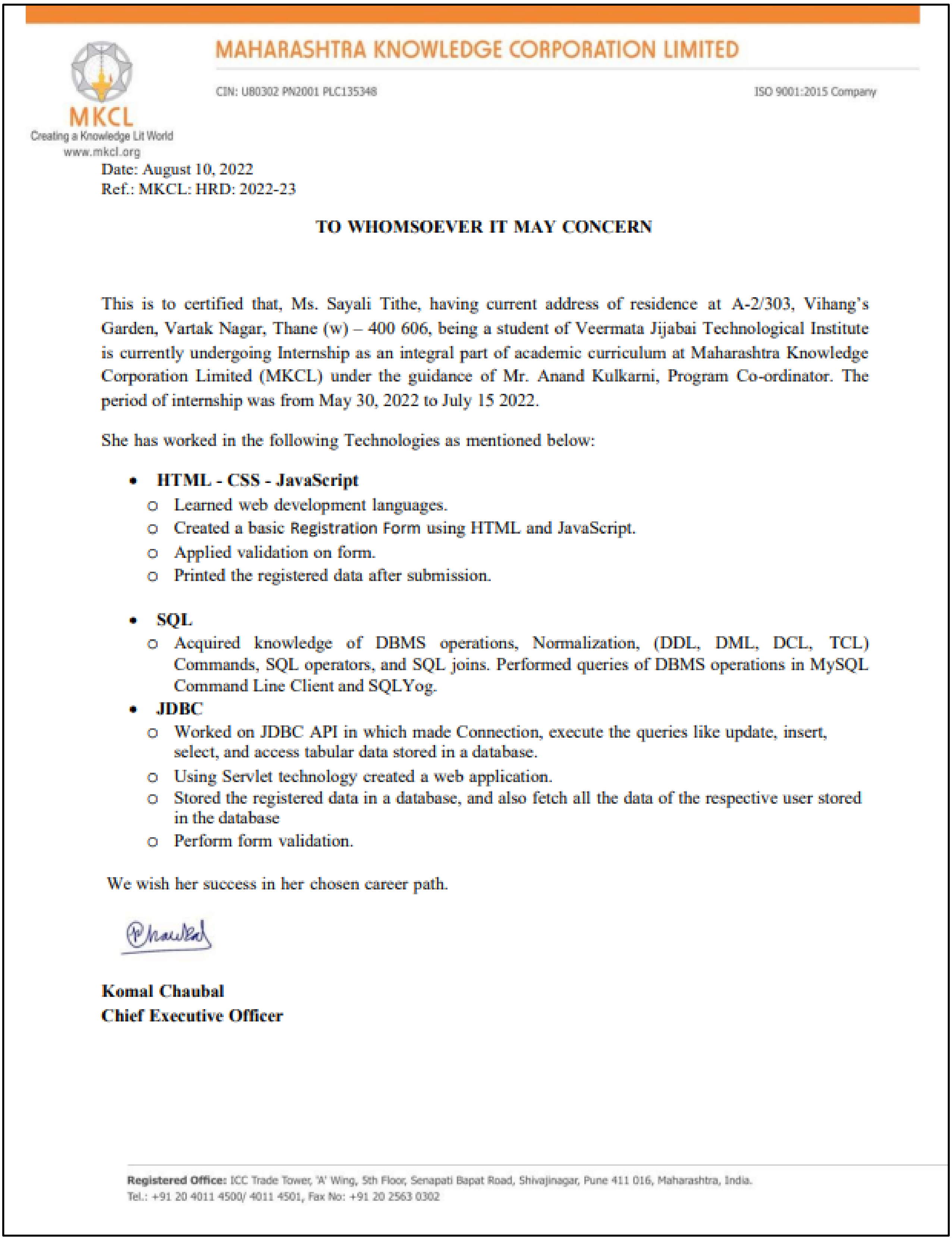
Examiner (Internship Supervisor)

Date :

Place : VJTI , Mumbai.

CONTENTS

INTERNSHIP CERTIFICATE



MAHARASHTRA KNOWLEDGE CORPORATION LIMITED

MKCL
Creating a Knowledge Lit World
www.mkcl.org

CIN: U80302 PN2001 PLC135348

ISO 9001:2015 Company

Date: August 10, 2022
Ref.: MKCL: HRD: 2022-23

TO WHOMSOEVER IT MAY CONCERN

This is to certified that, Ms. Sayali Tithe, having current address of residence at A-2/303, Vihang's Garden, Vartak Nagar, Thane (w) – 400 606, being a student of Veermata Jijabai Technological Institute is currently undergoing Internship as an integral part of academic curriculum at Maharashtra Knowledge Corporation Limited (MKCL) under the guidance of Mr. Anand Kulkarni, Program Co-ordinator. The period of internship was from May 30, 2022 to July 15 2022.

She has worked in the following Technologies as mentioned below:

- **HTML - CSS - JavaScript**
  - Learned web development languages.
  - Created a basic Registration Form using HTML and JavaScript.
  - Applied validation on form.
  - Printed the registered data after submission.
- **SQL**
  - Acquired knowledge of DBMS operations, Normalization, (DDL, DML, DCL, TCL) Commands, SQL operators, and SQL joins. Performed queries of DBMS operations in MySQL Command Line Client and SQLYog.
- **JDBC**
  - Worked on JDBC API in which made Connection, execute the queries like update, insert, select, and access tabular data stored in a database.
  - Using Servlet technology created a web application.
  - Stored the registered data in a database, and also fetch all the data of the respective user stored in the database
  - Perform form validation.

We wish her success in her chosen career path.

Komal Chaubal
Chief Executive Officer

Registered Office: ICC Trade Tower, 'A' Wing, 5th Floor, Senapati Bapat Road, Shivajinagar, Pune 411 016, Maharashtra, India.
Tel.: +91 20 4011 4500/ 4011 4501, Fax No: +91 20 2563 0302

ACKNOWLEDGMENT

I would like to express my gratitude to Veermata Jijabai Technological Institute, Matunga, Mumbai for including the internship program as a credit course which has provided an opportunity to gain practical working experience in the organization. I am thankful to all faculty members, for their constant encouragement and support.

A special thanks to the Maharashtra Knowledge Corporation Limited (MKCL) & MR. NATARAJ KATAKDHOND (Program Co-Ordinator) for giving me the opportunity to work as an Intern in their company.

I convey my sincere gratitude to MR. ANAND KULKARNI, Project Manager, and MISS. KOMAL AGRAWAL, Project Trainee helping me in my day-to-day activities during the intern time. I extend my grateful thanks to the teammates who help me with all necessary needs as well as shared their knowledge to improve my practical knowledge for the task.

This dissertation would not have been possible without the guidance and help of several individuals who in one way or another contributed and extended their valuable assistance in the preparation and completion of this internship.

CHAPTER 1: INTRODUCTION

1.1 ABSTRACT

In VJTI the 2-month Summer Internship Program in SEM 7 is a vital part of the curriculum. Since the majority of the students come without any prior work experience when they join after college, the Summer Internship adds worth to their CVs by giving each student immense learning. At VJTI, a student can bag an internship through various means since the institute gives us ample opportunities to interact with industry experts. I got my internship through the internet at MKCL.

After joining, My interaction at MKCL started with a small introduction to my college and my branch. As I belonged to the circuit branch, they asked me about my application and programming knowledge of mine. After our discussion, Mr. Anand Kulkarni, the Project Manager assigned me to Miss. Komal Agrawal, Project Trainee.

MKCL is a government organization that works on projects which contain official information and websites are confidential that can't be disclosed. They do not allow third-parties to interfere hence I was not able to become part of any project. They assigned me activities and tasks based on those activities which will help me to build my programming knowledge.

This report describes my internship experience at the Maharashtra Knowledge Corporation Limited (MKCL) in Navi Mumbai. Chapter Two summarizes the tasks that I completed which provide the background and purpose of the duties that were assigned to me. Chapters Three and four, included programming languages used and software used throughout the task completion. Chapter Five describes the process and techniques I used to compose the assigned tasks. At the end of this document, it explains how this task contributed to my development.

1.2 ABOUT COMPANY

Maharashtra Knowledge Corporation Limited (MKCL) was promoted by the Department of Higher and Technical Education, Government of Maharashtra, and shifted to the General Administration Department (GAD), Government of Maharashtra, India on January 05, 2018, and was incorporated under the Companies Act, 1956. Maharashtra Knowledge Corporation Limited (MKCL) was promoted by the Department of Higher and Technical Education (H & TE), Government of Maharashtra (GoM), India and was incorporated under the Companies Act, 1956. In order to make the capabilities of MKCL available to all the Departments of the Government of Maharashtra, governmental coordination of the affairs of the company has been handed over from January 05, 2018, to the General Administration Department (GAD) from the Department of Higher & Technical Education. Subsequently, it was allocated to the Directorate of Information Technology under GAD.

Creating a knowledge-led economy and knowledge-based societies in the world is key to the survival and development of people in the emerging knowledge era. MKCL believes that there exists a great potential in our society to emerge as a knowledge-led economy due to its large but latent talent pool of a young population. To develop learning, governance, and empowerment systems that are world-class and value-based and which are responsive to the individual and social developmental needs of the people by bridging the Digital Divide. MKCL commenced its business operations in April 2002 over the entire state of Maharashtra (the most industrialized state in India admeasuring about 300,000 sq. km. and having a population exceeding 100 million) and in last two decades it has emerged as a high-tech and high-touch initiative focused on design, development, and delivery of innovative eLearning, eGovernance, and eEmpowerment technologies, solutions, and services to its ever-growing base of millions of customers in public at large, Universities, Governments, and Communities.

CHAPTER 2: TASK ASSIGNED

We started with brushing up on basic programming concepts like searching, sorting and insertion using array & linked list.

2.1 REGISTRATION FORM

Create a register.html file with a basic HTML template. Next, inside the <head> tag, we can find the <title> tag. The content mentioned inside the <title> tag appears on the tab's name as it refers to the title of the current web page. So, change the text to Register. Now, if we run the HTML page in the browser, the HTML structure for the registration form Next, inside the <body> tag will appear on the screen. Our entire registration form will be created inside this <form> tag. Form tags are used to create any form in HTML.

Among many attributes, the two most important attributes in a form tag are action and method. The action attribute specifies where to send the form data once the user submits it, and it will be in the form of a URL, while the method attribute specifies the HTTP method which is used to send the form data, such as GET and POST. With the help of these all tags web page is formed.

2.2 VALIDATION OF FORM

HTML form validation can be done by JavaScript. If a form field (Name) is empty, this function alerts a message, and returns false, to prevent the form from being submitted. Data validation is the process of ensuring that user input is clean, correct, and useful.

Typical validation tasks are:

1. Has the user filled in all required fields?
2. Has the user entered a valid date?
3. Has the user entered text in a numeric field?

Most often, the purpose of data validation is to ensure correct user input. Validation can be defined by many different methods and deployed in many different ways. Server-side validation is performed by a web server after the input has been sent to the server. Client-side validation is performed by a web browser before the input is sent to a web server. To show the fields are not filled alert pop-up message is shown on the screen. Also, another option for validation is a paragraph tag of HTML that can be used and made a function in JS which will take values for each input field, will check its length is greater than zero or not and then shown error according to the condition.

2.3 STORAGE OF DATA

Create an entry HTML form capable of passing information to a secondary file. You must create the database and tables before passing data to them. The table's field names must match the names of the variables passed by the global variables. To store the data in the database you need to establish a connection between Html registration form and the database. For making connection using java we need to use JDBC. JDBC is used for connection between the front end i.e your Java Program and the back end i.e the database.

2.4 JDBC CONNECTION

Steps For Connectivity Between Java Program and Database

Step 1: Import the Packages

Step 2: Loading the drivers

In order to begin with, you first need to load the driver or register it before using it in the program. Registration is to be done once in your program. You can register a driver in one of two ways mentioned below as follows: 2-A Class.forName() - load the driver's class file into memory at the runtime. No need of using new or create objects. The Class.forName() uses to load the driver. 2-B DriverManager.registerDriver() - DriverManager is a Java inbuilt class with a static member register. Here we call the constructor of the driver class at compile time.

Step 3: Establish a connection using the Connection class object

After loading the driver, establish connections as shown below as follows:
Connection con = DriverManager.getConnection(url,user,password)
user: Username from which your SQL command prompt can be accessed. password: password from which the SQL command prompt can be accessed.
con: It is a reference to the Connection interface.
Url: Uniform Resource Locator which contains the information of database.
All 3 parameters above are of String type and are to be declared by the programmer before calling the function.

Step 4: Create a statement

Once a connection is established you can interact with the database. The JDBCStatement, CallableStatement, and PreparedStatement interfaces define the methods that enable you to send SQL commands and receive data from your database.

Step 5: Execute the query

The most important part i.e executing the query. The query here is an SQL Query. Now we know we can have multiple types of queries. Some of them are as follows: The query for updating/inserting a table in a database. The query for retrieving data. The executeQuery() method of the Statement interface is used to execute queries of retrieving values from the database. This method returns the object of ResultSet that can be used to get all the records of a table. The executeUpdate(sql query) method of the Statement interface is used to execute queries of updating/inserting.

Step 6: Closing the connections

So finally we have sent the data to the specified location and now we are on the verge of completing our task. By closing the connection, objects of Statement and ResultSet will be closed automatically. The close() method of the Connection interface is used to close the connection.

CHAPTER 3: PROGRAMMING LANGUAGES USED

3.1 HYPERTEXT TRANSFER PROTOCOL

HTML is a text file containing specific syntax, file and naming conventions that show the computer and the web server that it is in HTML and should be read as such. By applying these HTML conventions to a text file in virtually any text editor, a user can write and design a basic webpage, and then upload it to the internet. The most basic of HTML conventions is the inclusion of a document type declaration at the beginning of the text file. This always comes first in the document, because it is the piece that affirmatively informs a computer that this is an HTML file. The document header typically looks like this: <!DOCTYPE html>. It should always be written that way, without any content inside it or breaking it up. Any content that comes before this declaration will not be recognized as HTML by a computer.

3.2 JAVASCRIPT

JavaScript is a lightweight, interpreted programming language. It is designed for creating network-centric applications. It is complimentary to and integrated with Java. JavaScript is very easy to implement because it is integrated with HTML. It is open and cross-platform. Javascript is the most popular programming language in the world and that makes it a programmer's great choice. Once you learnt Javascript, it helps you developing great front-end as well as back-end softwares using different Javascript based frameworks like jQuery, Node.JS etc. Javascript is everywhere, it comes installed on every modern web browser and so to learn Javascript you really do not need any special environment setup. For example Chrome, Mozilla Firefox , Safari and every browser you know as of today, supports Javascript.

3.3 CASCADING STYLE SHEETS

CSS is used to control the style of a web document in a simple and easy way. CSS is the acronym for "Cascading Style Sheet". This tutorial covers both the versions CSS1,CSS2 and CSS3, and gives a complete understanding of CSS, starting from its basics to advanced concepts. Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable. CSS is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain.

3.4 JAVA

Java is an object-oriented, class-based, concurrent, secured and general-purpose computer-programming language. It is a widely used robust technology. Java is one of the most popular and widely used programming languages. Java has been one of the most popular programming languages for many years. Java is Object Oriented. However, it is not considered as pure object-oriented as it provides support for primitive data types (like int, char, etc) The Java codes are first compiled into byte code (machine-independent code). Then the byte code runs on Java Virtual Machine (JVM) regardless of the underlying architecture. Java syntax is similar to C/C++. But Java does not provide low-level programming functionalities like pointers. Also, Java codes are always written in the form of classes and objects. Java is used in all kinds of applications like Mobile Applications (Android is Java-based), desktop applications, web applications, client-server applications, enterprise applications, and many more.

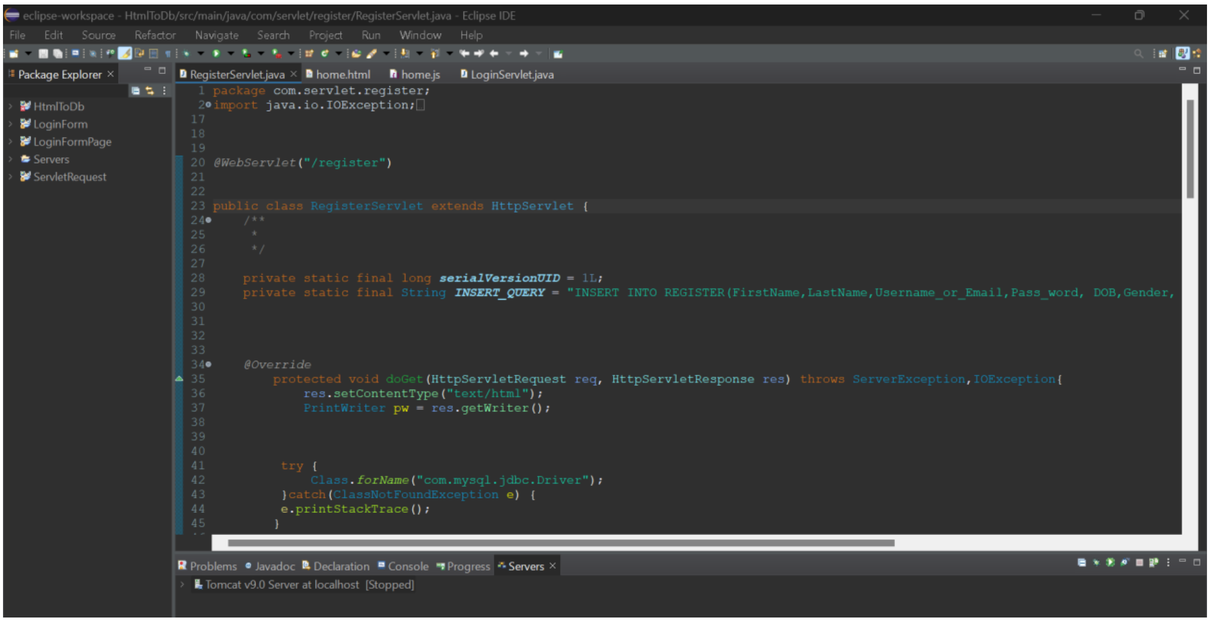
3.4 STRUCTURED QUERY LANGUAGE

SQL (pronounced “ess-que-el”) stands for Structured Query Language. SQL is used to communicate with a database. According to ANSI (American National Standards Institute), it is the standard language for relational database management systems. SQL statements are used to perform tasks such as update data on a database, or retrieve data from a database. Some common relational database management systems that use SQL are: Oracle, Sybase, Microsoft SQL Server, Access, Ingres, etc. Although most database systems use SQL, most of them also have their own additional proprietary extensions that are usually only used on their system. However, the standard SQL commands such as “Select”, “Insert”, “Update”, “Delete”, “Create”, and “Drop” can be used to accomplish almost everything that one needs to do with a database. This tutorial will provide you with the instruction on the basics of each of these commands as well as allow you to put them to practice using the SQL Interpreter.

CHAPTER 4: SOFTWARE USED

4.1 ECLIPSE

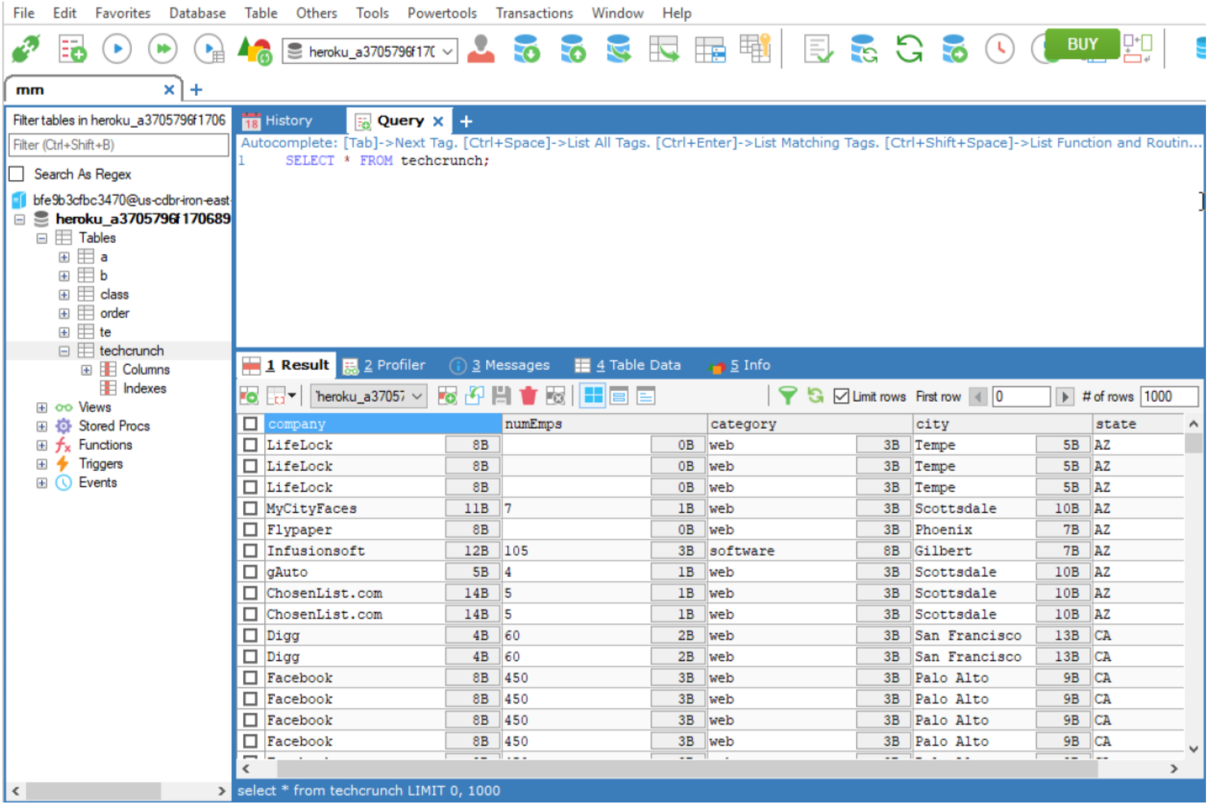
Eclipse is an integrated development environment (IDE) for developing applications using the Java programming language and other programming languages such as C/C++, Pythn, PERL, Ruby etc. The Eclipse platform which provides the foundation for the Eclipse IDE is composed of plug-ins and is designed to be extensible using additional plug-ins. Developed using Java, the Eclipse platform can be used to develop rich client applications, integrated development environments and other tools. Eclipse can be used as an IDE for any programming language for which a plug-in is available. The Eclipse Java IDE provides many debugging tools and views grouped in the Debug Perspective to help the you as a developer debug effectively and efficiently. There are many improvements included in the latest Eclipse Java Development Tools (JDT) release included in the Eclipse Oxygen Simultaneous Release.


eclipse-workspace - HtmlToDb/src/main/java/com/servlet/register/RegisterServlet.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
HtmlToDb
LoginForm
LoginFormPage
Servers
ServletRequest
RegisterServlet.java
home.html
home.js
LoginServlet.java
Problems Javadoc Declaration Console Progress Servers
Tomcat v9.0 Server at localhost [Stopped]

```
package com.servlet.register;
import java.io.IOException;



@WebServlet("/register")


public class RegisterServlet extends HttpServlet {
    /**
     *
     */

    private static final long serialVersionUID = 1L;
    private static final String INSERT_QUERY = "INSERT INTO REGISTER(FirstName,LastName,Username_or_Email,Pass_word, DOB,Gender,




    @Override
        protected void doGet(HttpServletRequest req, HttpServletResponse res) throws ServerException,IOException{
            res.setContentType("text/html");
            PrintWriter pw = res.getWriter();



          try {
              Class.forName("com.mysql.jdbc.Driver");
          }catch(ClassNotFoundException e) {
          e.printStackTrace();
          }
```

4.2 SQLYOG

SQLyog Ultimate is the most powerful manager, admin and GUI tool for MySQL, combining the features of MySQL Query Browser, Administrator, phpMyAdmin and other MySQL Front Ends and MySQL GUI tools in a single intuitive interface. SQLyog is a fast, easy to use and compact graphical tool for managing your MySQL databases. SQLyog Ultimate was developed for all who use MySQL as their preferred RDBMS. Whether you enjoy the control of handwritten SQL or prefer to work in a visual environment, SQLyog makes it easy for you to get started and provides you with tools to enhance your MySQL experience. MySQL manager and admin tool SQLyog Ultimate provides you with powerful means to manage your MySQL databases. It features the simplicity of MySQL Front, with the power of EMS MySQL Admin. SQLyog provides detailed profile information for every SQL statement executed. Session Restore SQLyog Ultimate is a powerful MySQL manager that restores your previous session the way you left it. Be it a system crash or accidentally closing your MySQL client. Schema and Data Sync Find and fix schema mismatching while syncing data from one database to another. Replicate data periodically to a different server at scheduled intervals. Compressed backups with scheduling Find the plethora of options in mysqldump daunting. Just back up with a single click interface in SQLyog MySQL GUI tool. Autocomplete and SQL formatting Save hours of typing. Write queries 10x faster.



Autocomplete: [Tab]->Next Tag. [Ctrl+Space]->List All Tags. [Ctrl+Enter]->List Matching Tags. [Ctrl+Shift+Space]->List Function and Routin...

```
SELECT * FROM techcrunch;
```

□	company		numEmps		category		city		state
□	LifeLock	8B		0B	web	3B	Tempe	5B	AZ
□	LifeLock	8B		0B	web	3B	Tempe	5B	AZ
□	LifeLock	8B		0B	web	3B	Tempe	5B	AZ
□	MyCityFaces	11B	7	1B	web	3B	Scottsdale	10B	AZ
□	Flypaper	8B		0B	web	3B	Phoenix	7B	AZ
□	Infusionsoft	12B	105	3B	software	8B	Gilbert	7B	AZ
□	gAuto	5B	4	1B	web	3B	Scottsdale	10B	AZ
□	ChosenList.com	14B	5	1B	web	3B	Scottsdale	10B	AZ
□	ChosenList.com	14B	5	1B	web	3B	Scottsdale	10B	AZ
□	Digg	4B	60	2B	web	3B	San Francisco	13B	CA
□	Digg	4B	60	2B	web	3B	San Francisco	13B	CA
□	Facebook	8B	450	3B	web	3B	Palo Alto	9B	CA
□	Facebook	8B	450	3B	web	3B	Palo Alto	9B	CA
□	Facebook	8B	450	3B	web	3B	Palo Alto	9B	CA
□	Facebook	8B	450	3B	web	3B	Palo Alto	9B	CA

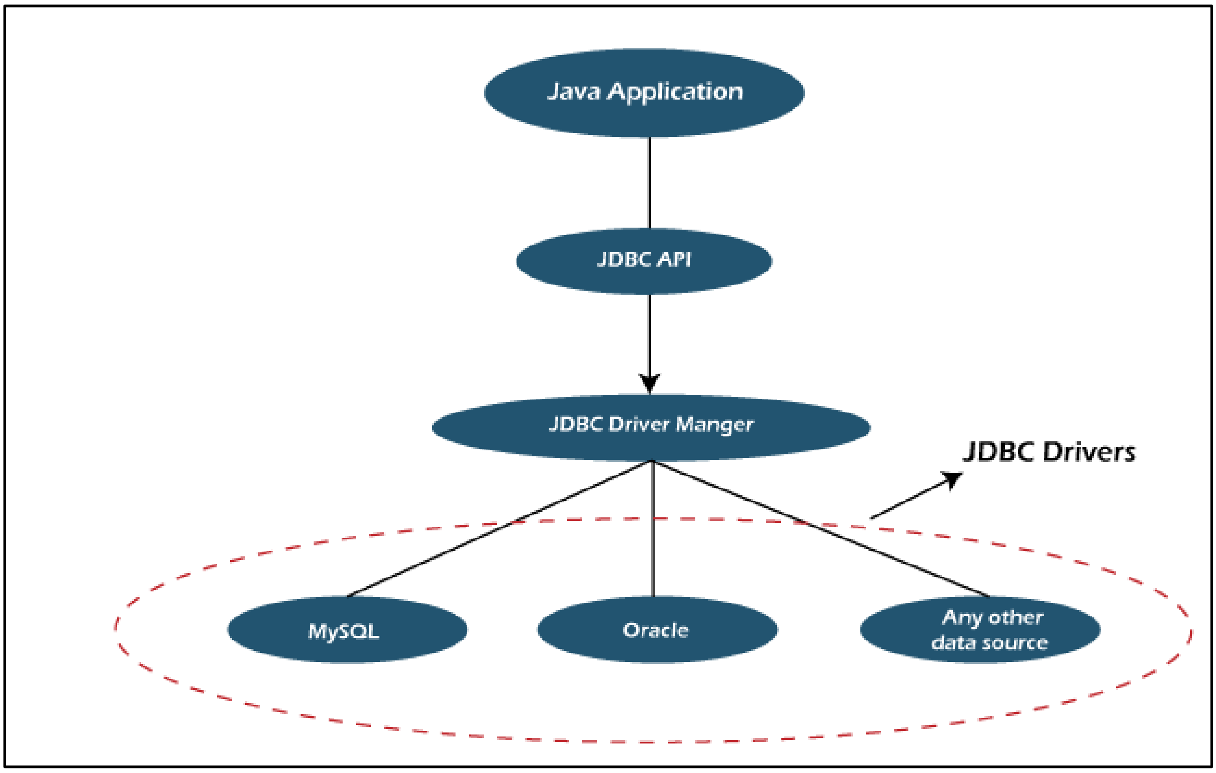
select * from techcrunch LIMIT 0, 1000

CHAPTER 5: TECHNOLOGIES USED

5.1 JAVA DATABASE CONNECTIVITY

JDBC or Java Database Connectivity is a Java API to connect and execute the query with the database. It is a specification from Sun microsystems that provides a standard abstraction(API or Protocol) for java applications to communicate with various databases. It provides the language with java database connectivity standards. It is used to write programs required to access databases. JDBC, along with the database driver, can access databases and spreadsheets. JDBC(Java Database Connectivity) is an API(Application programming interface) used in java programming to interact with databases. The classes and interfaces of JDBC allow the application to send requests made by users to the specified database. Purpose of JDBC Enterprise applications created using the JAVA EE technology need to interact with databases to store application-specific information. So, interacting with a database requires efficient database connectivity, which can be achieved by using the ODBC(Open database connectivity) driver. Components of JDBC There are generally four main components of JDBC through which it can interact with a database.

Architecture of JDBC



Architecture of JDBC Description:

1. JDBC API: It provides various methods and interfaces for easy communication with the database. It provides two packages as follows, which contain the java SE and Java EE platforms to exhibit WORA(write once run anywhere) capabilities. java.sql.*; It also provides a standard to connect a database to a client application.
2. JDBC Driver manager: It loads a database-specific driver in an application to establish a connection with a database. It is used to make a database-specific call to the database to process the user request.
3. JDBC Test suite: It is used to test the operation(such as insertion, deletion, updation) being performed by JDBC Drivers.
4. JDBC-ODBC Bridge Drivers: It connects database drivers to the database. This bridge translates the JDBC method call to the ODBC function call. It makes use of the sun.jdbc.odbc package which includes a native library to access ODBC characteristics.

Application: It is a java applet or a servlet that communicates with a data source. The JDBC API: The JDBC API allows Java programs to execute SQL statements and retrieve results. Some of the important classes and interfaces defined in JDBC API are as follows:

DriverManager: It plays an important role in the JDBC architecture. It uses some database-specific drivers to effectively connect enterprise applications to databases.

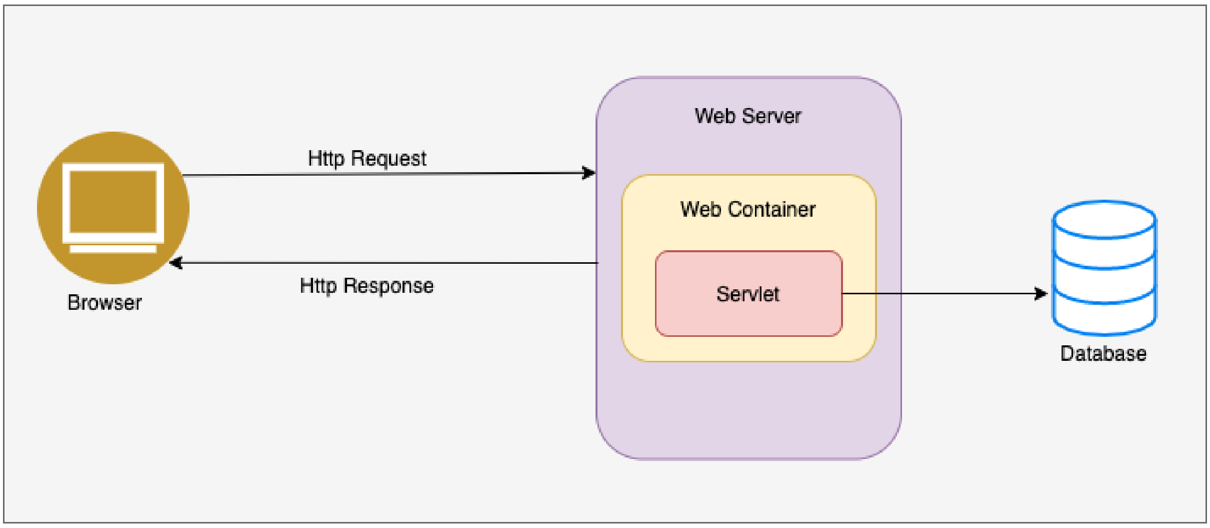
JDBC drivers: To communicate with a data source through JDBC, you need a JDBC driver that intelligently communicates with the respective data source.

5.2 JAVA SERVLETS

Java Servlets are software component that runs on a web server or an application server and be responsible to receive the request from the web server, process the request, and respond back to the server. Servlets extends the capabilities of a server as they can respond to many types of requests, they act like web containers for hosting web applications on web servers. They support nearly all the client–server protocols, but is often used with HTTP and also known as HTTP servlet.

- The user sends HTTP requests to the web server
- The server has the web container containing servlet, which gathers data from the database and creates a response.
- The response created by servlet is sent through HTTP Response to the client browser. As the web server works purely on the HTTP protocol. Hence, this

conversion from servlet's response to HTTP response is taken care by the web container.



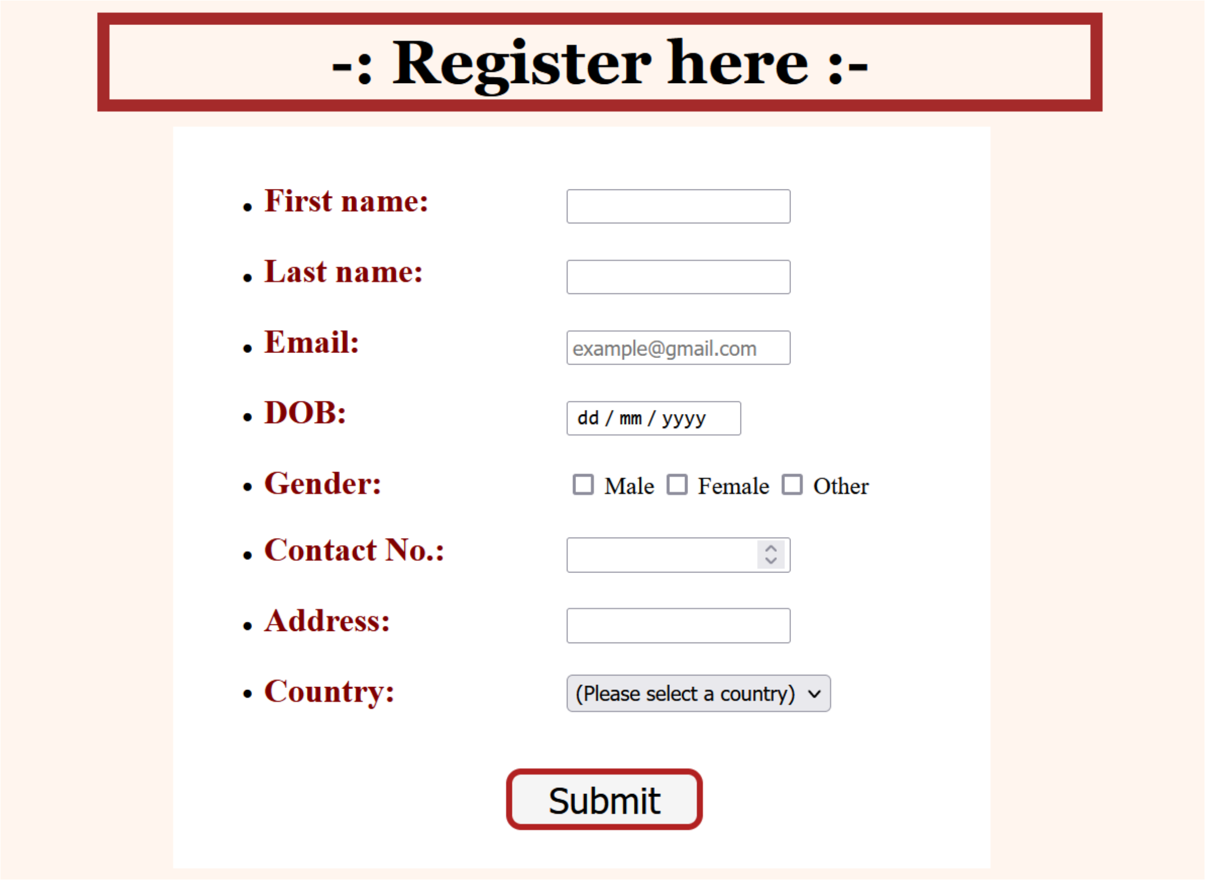
Web container

Web container is also known as servlet container or servlet engine. It provides the runtime environment for Java EE (j2ee) applications. The client/user can request only a static web page from the server. If the user wants to read the web pages as per input then the web container is used in java. Therefore, it implements the Servlet API and the services required to process HTTP requests. Web container initiates the Servlet that matches the requested URL by invoking the service() method of Servlet class. The service() method which is invoked for a given HTTP request is handled in a separate thread within the web container protocol.

- The User sends an HTTP Request to the web server
- Web server forwards requests to Web Container
- Web Container forwards the request to the Servlet in form of the request object
- Servlet builds the response object and sends it back to the Web Container
- Web container transforms the response object to equivalent HTTP response and sends it to the web server
- The web server sends the response via HTTP response back to the client.

CHAPTER 6: CODES & RESULT

Registration form:



-: Register here :-

- First name:
- Last name:
- Email: example@gmail.com
- DOB: dd / mm / yyyy
- Gender: □ Male □ Female □ Other
- Contact No.:
- Address:
- Country: (Please select a country)

Java Code: Code that stores the data from the registration form to SQL.

```
package com.servlet.register;
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.*;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.rmi.ServerException;

@WebServlet("/register")

public class RegisterServlet extends HttpServlet {
    /**
      *
      */

private static final long serialVersionUID = 1L;
private static final String INSERT_QUERY = "INSERT INTO
REGISTER(FirstName,LastName,Username_or_Email,Pass_word, DOB,Gender,
Contact, Place, Country) VALUES(?,?,?,?,?,?,?,?,?)";
```

```
@Override
protected void doGet(HttpServletRequest req, HttpServletResponse res)
throws ServerException,IOException{
      res.setContentType("text/html");
      PrintWriter pw = res.getWriter();


try {
    Class.forName("com.mysql.jdbc.Driver");
    }catch(ClassNotFoundException e) {
    e.printStackTrace();}


try(Connection con =
DriverManager.getConnection("jdbc:mysql:///learn","root","S@y@l!123");
PreparedStatement ps = con.prepareStatement(INSERT_QUERY);){


               String firstName = req.getParameter("fname");
               String lastName = req.getParameter("lname");
               String Username_or_Email = req.getParameter("Email");
               String Pass_word = req.getParameter("Password");
               String DOB = req.getParameter("Date");
               String Gender=" ";
               String[] choose=req.getParameterValues("gender_type");
                 for(String i:choose){
                        Gender=i;
                 }

               String Contact = req.getParameter("contact_number");
               String Place = req.getParameter("place");
               String Country = req.getParameter("country");

                  ps.setString(1,firstName);
                  ps.setString(2,lastName);
                  ps.setString(3,Username_or_Email);
                  ps.setString(4,Pass_word);
                  ps.setString(5,DOB);
                  ps.setString(6,Gender);
                  ps.setString(7,Contact);
                  ps.setString(8,Place);
                  ps.setString(9,Country);


                   int count= ps.executeUpdate();
                         if(count==0 ) {
                        pw.println("Record is not stored into database");
                  }else {
                        pw.println("Record is stored into database");
                  }


           }catch(SQLException se) {
              pw.println(se.getMessage());
              se.printStackTrace();
           }catch(Exception e) {
               pw.println(e.getMessage());
               e.printStackTrace();
           }
            pw.close();
        }
```

```
 @Override
 protected void doPost (HttpServletRequest req, HttpServletResponse res)
throws ServerException ,IOException{
            doGet(req,res);
        }
}
```

HTML Code: Code to build the registration form.

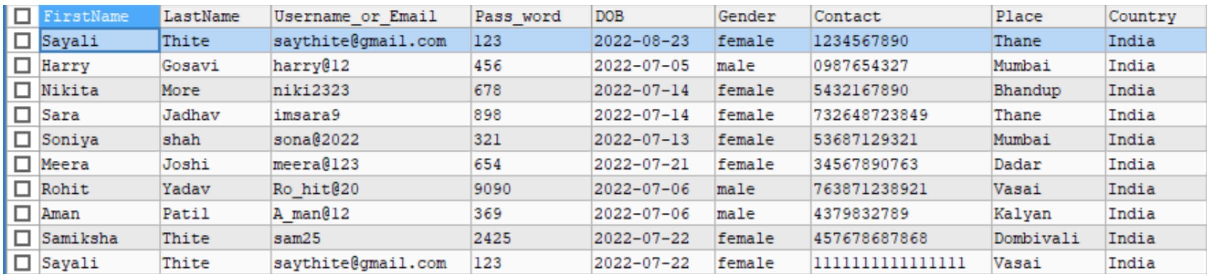
```
<!DOCTYPE html>
<html lang="en">

<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Registration Form</title>
    <style>
        body {background-color: #fff5ee;}

        h1 {
            text-align: center;
              font-size: 40px;
              margin-top: 10px ;
                  margin-right: 300px ;
                  margin-bottom: 10px  ;
                  margin-left: 300px ;
                padding: 2px;
                  border: 8px solid #a52a2a;
                  font-family: Georgia, serif;}

        button {
            background-color: #f5f5f5;
            border-radius: 10px;
            border: 4px solid       #b22222;
            color:      #000000;
            text-align: center;
            font-size: 24px;
            padding: 3px;
            width: 130px;
            margin-left: 150px;
            margin-bottom:10px;}

        label {
            color: #800000;
            margin-bottom:10px;
            font-size: 22px;
            font-weight: bold;
            width: 200px;
            display: inline-block;
                float: left;
                clear: left;
                text-align: left;
            left:100px;
            margin-top:3px;
                  }

        input{
            height:17px;
            margin-top:3px;
            }

        .TextError {
            padding:-30px;
            margin-top:-15px;
            margin-bottom:15px;
            margin-left:80px;
            color:#c12020;
        }

        .SelectError{
            color:#c12020;
            margin-top:-30px;
            margin-bottom:30px;
            margin-left:200px;
        }

        .Container{
             max-width: 500px;
             width: 100%;
             background:#ffffff;
             margin-left:350px;
             height: 500px;
             padding: 18px; }


            select{
            height:25px;
            margin-top:3px;
            }

    </style>
</head>
<body >

    <div class ="form">
        <form action="register" method="get" name="RegisterForm"
onsubmit="return checkValidation(this)">
            <!-- <form method ="post" name="RegisterForm" onsubmit="return
checkforblank()">   -->
    <h1>-: Register here :-</h1>
    </div>

    <div class="Container">
    <ul>

    <li><label for="first_name">First name: </label>
    <input type="text" name="fname" id ="first_name" size="20"
/><br><br></li>
    <p class="TextError" id="fname_error"></p>


    <li> <label for="last_name">Last name: </label>
    <input type="text" name="lname" id ="last_name" size="20"
><br><br></li>
    <p class="TextError" id="lname_error"></p>

    <li> <label for="email">Username or Email: </label>
    <input type="text" name="Email" id ="email" size="20"  /></li><br>
    <p class="TextError" id="Email_error"></p>
    <li> <label for="Password_register">Password: </label>
    <input type="text" name="Password" id ="Password_register" size="20"
/></li><br>
    <p class="TextError" id="Password_error"></p>

    <li><label for="birth_date" >DOB: </label>
    <input type="date"  name="Date" id="birth_date" ></li> <br>
    <p class="TextError" id="Date_error"></p>

    <li><label for="gender" name="gender_type">Gender: </label>
    <input type="checkbox" name ="gender_type" value ="male"
id="check_male"/>
    <text for="check1">Male </text>
    <input type="checkbox" name ="gender_type" value ="female"
id="check_female" />
    <text for="check2">Female</text>
    <input type="checkbox" name ="gender_type" value ="other"
id="check_other" />
    <text for="check3">Other</text><br><br></li>
    <p class="TextError" id="gender_error"></p>

    <li><label for="contact">Contact No.: </label>
    <input type="number" name="contact_number" id ="contact" size="20"
/><br><br></li>
    <p class="TextError" id="contact_error"></p>

    <li><label for="address">Address: </label>
    <input type="text" name="place" id ="address" size="20" /></li><br>
    <p class="TextError" id="place_error"></p>


     <li><label for="selected" >Country:</label>
        <select name="country" id ="selected">
        <option id="sel" value="(Please select a country)">(Please select a
country)</option>
        <option id ="sel1" value="Australia">Australia</option>
        <option id ="sel2" value="Canada">Canada</option>
        <option id ="sel3" value="India">India</option>
        <option id ="sel4" value="Russia">Russia</option>
        <option id ="sel5"  value="USA">USA</option></select></li><br><br>
    <p class="SelectError" id="selected_error"></p>

    <button id="submit_button" class="button" name="submit_button"
>Submit</button>

</ul>
</div>
    </form>

    <script src="home.js"></script>

   </body>
</html>
```

Javascript Code: Code to check the validation of each input field.

```
function checkValidation (){
var nameCheck = document.getElementById("first_name").value;
    if(nameCheck<=0){
```

```
    document.getElementById("fname_error").innerHTML = '*Please enter your 
first name*';
            return false;}
    else{document.getElementById("fname_error").innerHTML = '';}

var lastCheck = document.getElementById("last_name").value;
    if(lastCheck<=0){
    document.getElementById("lname_error").innerHTML = '*Please enter your 
last name*';
            return false;}
    else{document.getElementById("lname_error").innerHTML = '';}

var emailCheck = document.getElementById("email").value;
    if(emailCheck<=0){
    document.getElementById("Email_error").innerHTML = '*Please enter your 
username or email*';
            return false;}
    else{document.getElementById("Email_error").innerHTML = '';}

var passwordCheck = document.getElementById("Password_register").value;
    if(passwordCheck<=0){
        document.getElementById("Password_error").innerHTML = '*Please 
enter your Password*';
        return false;}
    else{document.getElementById("Password_error").innerHTML = '';

   }

var DOBCheck = document.getElementById("birth_date").value;
    if(DOBCheck<=0){
    document.getElementById("Date_error").innerHTML = '*Please enter your 
date of birth*';
            return false;}
    else{document.getElementById("Date_error").innerHTML = '';}

var check_maleBox = document.getElementById("check_male");
var check_femaleBox= document.getElementById("check_female");
var check_otherBox = document.getElementById("check_other");

if (check_maleBox.checked == false && check_femaleBox.checked == false && 
check_otherBox.checked == false) {
    document.getElementById("gender_error").innerHTML = '*Please select 
your gender*';
    return false;}
else{ document.getElementById("gender_error").innerHTML = '';}


var contactCheck = document.getElementById("contact").value;
    if(contactCheck.length <= 0 || contactCheck.length <= 9){
    document.getElementById("contact_error").innerHTML = '*Please enter 
your contact*';
            return false;}
    else{document.getElementById("contact_error").innerHTML = '';}

var addressCheck = document.getElementById("address").value;
    if(addressCheck<=0){
     console.log(addressCheck);
    document.getElementById("place_error").innerHTML = '*Please enter your 
address*';
            return false;}
    else{document.getElementById("place_error").innerHTML = '';
    console.log(addressCheck);
    }

var countryCheck = document.getElementById("selected").value;
    if(countryCheck == "(Please select a country)"){

        document.getElementById("selected_error").innerHTML = '*Please
enter your Country*';
        return false;}
    else{document.getElementById("selected_error").innerHTML = '';

   }

    }
```

All the values get stored in the database when the user clicks on submit button.



☐	FirstName	LastName	Username_or_Email	Pass_word	DOB	Gender	Contact	Place	Country
☐	Sayali	Thite	saythite@gmail.com	123	2022-08-23	female	1234567890	Thane	India
☐	Harry	Gosavi	harry@12	456	2022-07-05	male	0987654327	Mumbai	India
☐	Nikita	More	niki2323	678	2022-07-14	female	5432167890	Bhandup	India
☐	Sara	Jadhav	imsara9	898	2022-07-14	female	732648723849	Thane	India
☐	Soniya	shah	sona@2022	321	2022-07-13	female	53687129321	Mumbai	India
☐	Meera	Joshi	meera@123	654	2022-07-21	female	34567890763	Dadar	India
☐	Rohit	Yadav	Ro_hit@20	9090	2022-07-06	male	763871238921	Vasai	India
☐	Aman	Patil	A_man@12	369	2022-07-06	male	4379832789	Kalyan	India
☐	Samiksha	Thite	sam25	2425	2022-07-22	female	457678687868	Dombivali	India
☐	Sayali	Thite	saythite@gmail.com	123	2022-07-22	female	1111111111111111	Vasai	India

CHAPTER 7: LEARNING EXPERIENCE

It was a great experience to work on such interesting technologies. Fully satisfied after performing such challenging and amazing tasks. I got an opportunity to work in the real-time tech world in which I learned about HTML, database management, and JDBC. Throughout the time, my learning graph increased exponentially. During the internship training, I found that several things are important:

Critical and Analytical Thinking

To organize our tasks and assignment, we need to analyze our problems and assignment and formulate a good solution to the problem. I learned to set a contingency plan for the solution so that we are well prepared for unforeseeable situations.

Time Management

During this internship period, I learned about proper time management. Effective time management allows me to do my assignment efficiently and meet my target which was assigned. I schedule all the tasks to avoid time wastage which allows me to plan ahead, and gain more as a result.

Colleague Interactions

In a working environment, teamwork is vital in contributing to a strong organization. Teamwork is also essential in reaching goals. As I was working on these technologies for the first time I interacted with experts and other teammates and solve errors and difficulties. Thus, communicating and sharing help me to complete the task.

CHAPTER 8: CONCLUSION

This internship has been an excellent and rewarding experience. I can conclude that there has been a lot I've learned from my work at MKCL. It was very productive and helped me to optimize my technical expertise. As someone with no prior experience with web development, I believe my time spent learning and implementing my knowledge step by step, was well worth it. It contributed to finding an acceptable solution to build a fully functional web page. Two main things that I've learned are the importance of time-management skills and self-motivation. On the whole, this internship was a useful experience. I have gained new knowledge, and skills and met many new people. I achieved several of my learning goals. The internship was also good to find out what my strengths and weaknesses are. This helped me to define what skills and knowledge I have to improve in the coming time. At last, this software-based internship experience allowed me to discover my interests and choose my specialization with a clear direction.