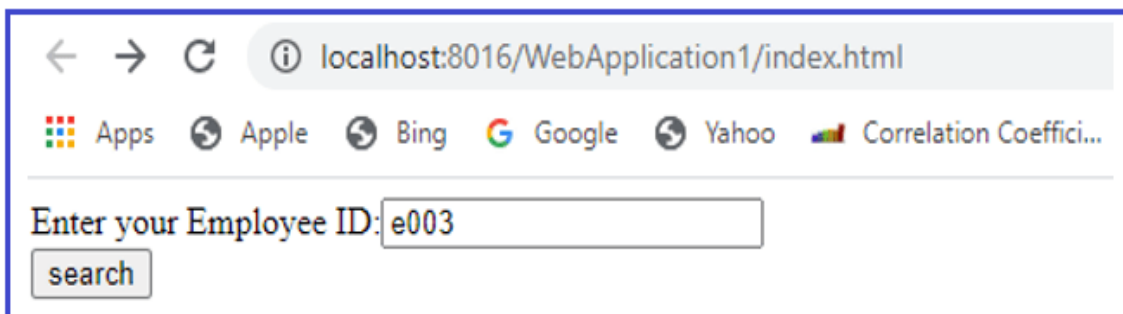


## How to Fetch Data from the Database using Servlet.

Here, we are using the below employee table, and based on the empid we will fetch the data from this employee table.(Search employee record using emp id)

atabase-using-servlet/

Enter Employee ID (empid) and click on the search button.

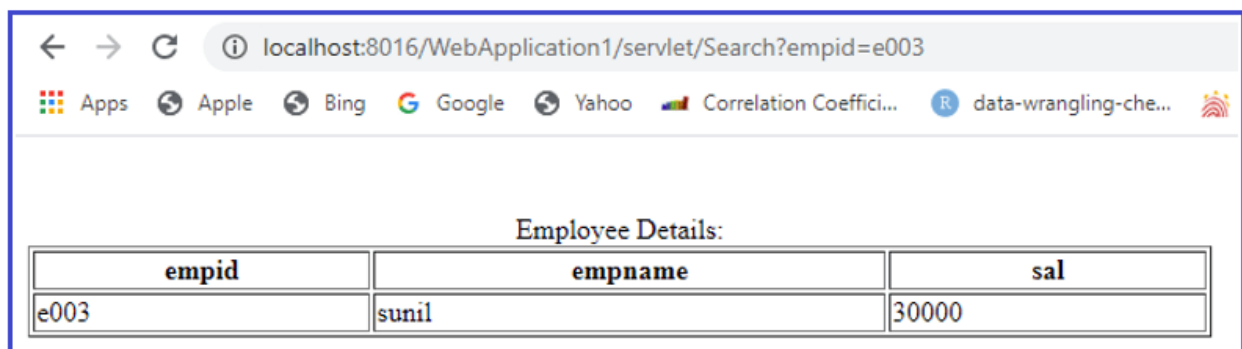


← → ↻ ⓘ localhost:8016/WebApplication1/index.html

Apps Apple Bing Google Yahoo Correlation Coeffici...

Enter your Employee ID:

After clicking on the search button, you will get the following output:



← → ↻ ⓘ localhost:8016/WebApplication1/servlet/Search?empid=e003

Apps Apple Bing Google Yahoo Correlation Coeffici... data-wrangling-che...

Employee Details:

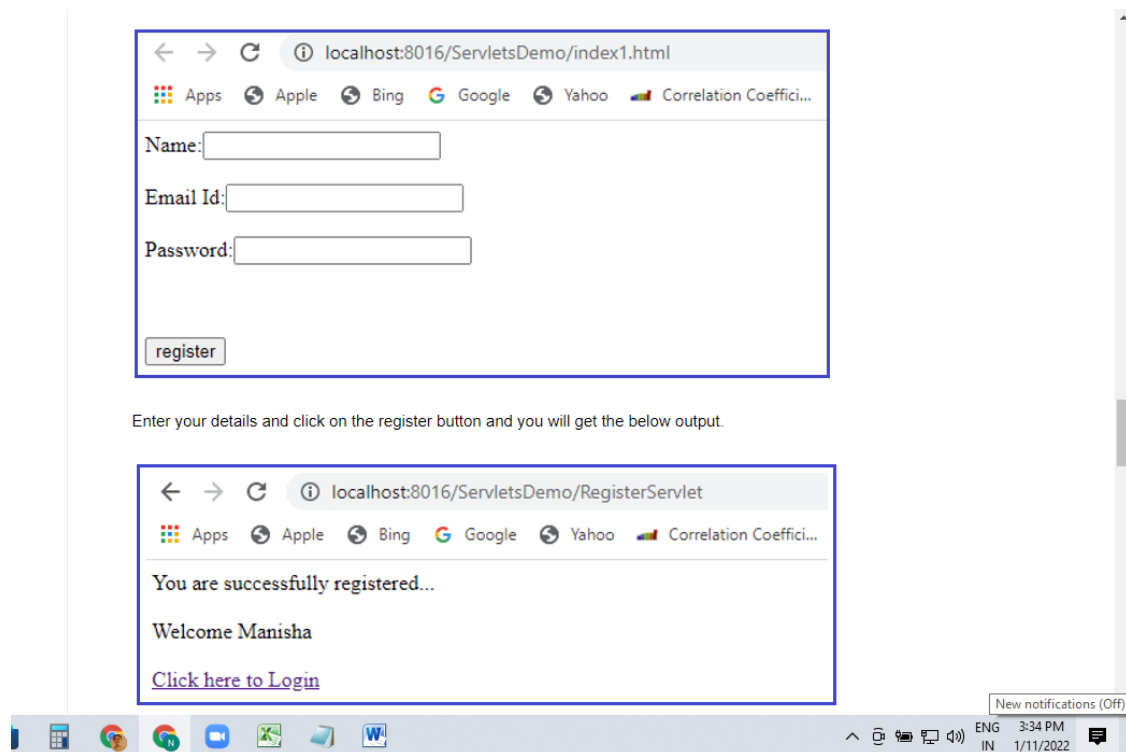
empid	empname	sal
e003	sunil	30000

## Develop a Registration Form in Servlet. To develop a registration form we will need to connect our servlet application with the database

Create three files:

1. **index1.html**
2. **RegisterServlet.java**
3. **Welcome1.java**

In index1.html, we have to get input from the user using text fields and a combo box. The information entered by the user has forwarded to RegisterServlet.java servlet which is responsible to store the data into the database. RegisterServlet.java servlet receives all the data entered by the user and stores it into the database. Here, we are performing the database logic. But you may separate it, which will be better for the web application.



**How to send Email through JavaMail API in Servlet** . For TLS and SSL authentication, we are going to use the Email SMTP server because it supports both. JavaMail API is not a part of standard JDK, so we have to download it from its official website. Download the latest version of the JavaMail reference implementation and include it in your project build path. The jar file name will be:

1. **javax.mail.jar**
2. **javax.activation.jar**

The example contains the following steps:

1. Creating **javax.mail.Session** object
2. Creating **javax.mail.internet.MimeMessage** object, we need to set different properties in this object such as recipient email address, Email Subject, Reply-To email, email body, etc.
3. Using **javax.mail.Transport** to send the email message.

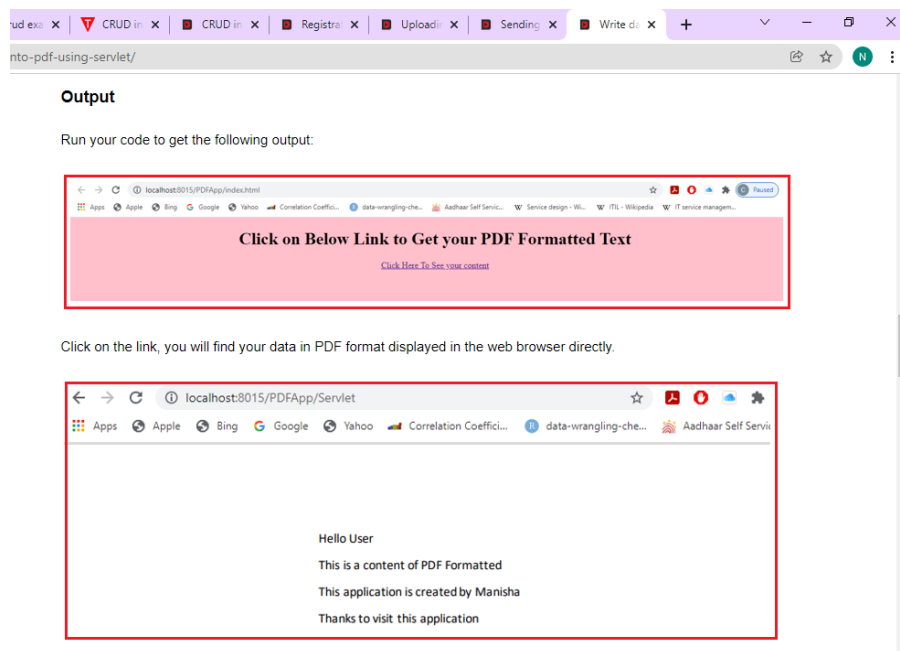
The logic to create a session differs based on the type of SMTP server, for example, if the SMTP server doesn't require any authentication, we can create the Session object with some simple properties whereas if the SMTP server requires TLS or SSL authentication, then the logic to create will differ. Here, I am setting some header properties in the **MimeMessage**, which are used by the email clients to properly render and display the email message. Here, I am using **Session.getInstance()** to get the Session object by passing the Properties object. We need to set the **mail.smtp.host** property with the SMTP server host. If the SMTP server is not running on the default port (25), then we need to set **mail.smtp.port** property.

**How to Write data into PDF using Servlet.** we can create an application that writes our data in PDF file format. We are writing some data in PDF using a servlet program and it will be displayed in the PDF file format. To create this application, you need to create the following files:

1. **HTML file (index.html)**
2. **servlet file (Servlet.java)**
3. **web.xml file**

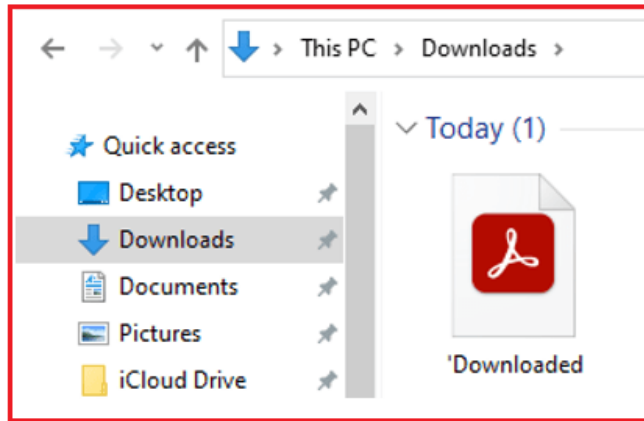
**index.html** file provides a link to the servlet through which our PDF content is displayed. **Servlet.java** file writes data as PDF and provides the information to the server that it is a PDF file type. **web.xml** is used to configure the servlet file and it provides servlet information to the server.

**Note:** We need to add the “spdf.jar” jar file for this application. Download this file directly from the Oracle website.



---

If you want to see your page format then you just save this page and see it in PDF format.



Open this downloaded file, you will get the file in PDF format as shown below.

