MongoDbServlet

Day 3 Work

1- Perform CRUD operation
2 - Project Work

Story 1 :Perform CRUD operation

Task 1: For performing Create operation

```
import com.mongodb.MongoClient;
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoDatabase;
import org.bson.*;
import org.bson.Document;
import java.util.Date;
import org.bson.conversions.Bson;
import com.mongodb.BasicDBObject;
import com.mongodb.DB;
import com.mongodb.DBCollection;
import java.io.*;
import java.util.*;
import com.mongodb.*;
public class MongoDemo {
public static void main(String[] args){
try{
//---------- Connecting DataBase ------------------------//
MongoClient mongoClient = new MongoClient( "localhost" , 27017 );
//---------- Creating DataBase ------------------------//
MongoDatabase db = mongoClient.getDatabase("kls");
//---------- Creating Collection ------------------------//
MongoCollection<Document> table = db.getCollection("employee");
//---------- Creating Document ------------------------//
Document doc = new Document("name", "Saurabh”);
doc.append("id",12);
//---------- Inserting Data ------------------------//
table.insertOne(doc);
```

```
}catch(Exception e){
System.out.println(e);
}
} }
```

Task 2 : For Delete Operation

Here first we insert document and after that we delete that document after insertion of document Document Inserted message printed on console and after del of db message also be shown on console

```
import com.mongodb.MongoClient;
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoDatabase;
import org.bson.*;
import org.bson.Document;
import java.util.Date;
import org.bson.conversions.Bson;
import com.mongodb.BasicDBObject;
import com.mongodb.DB;
import com.mongodb.DBCollection;
import java.io.*;
import java.util.*;
import com.mongodb.*;
public class MongoDemo {
public static void main(String[] args){
try{
//---------- Connecting DataBase -------------------------//
MongoClient mongoClient = new MongoClient( "localhost" , 27017 );
//---------- Creating DataBase ---------------------------//
MongoDatabase db = mongoClient.getDatabase("kls");
//---------- Creating Collection -------------------------//
```

```
MongoCollection<Document> table = db.getCollection("employee");
//---------- Creating Document ----------------------------//
           DBObject document1 = new BasicDBObject();
              document1.put("name", "Saurabh");
              document1.put("pass", "Saurabh");
              document1.put("mobile Number","9648982222");
              document1.put("email","saurabhtvns@gmail.com");
              collection.insert(document1);
              System.out.println("Document Inserted");
              collection.remove(document1);
              System.out.println("Document Deleted");
}}}
```

Task 3: For performing update operation

In this we give first id because update record based on the id in this document we update email so first we set id of that field which we want to update and after that we give new value $set used to set the new value

```
            import com.mongodb.MongoClient;
            import com.mongodb.client.MongoCollection;
            import com.mongodb.client.MongoDatabase;
            import org.bson.*;
            import org.bson.Document;
            import java.util.Date;
            import org.bson.conversions.Bson;
            import com.mongodb.BasicDBObject;
            import com.mongodb.DB;
            import com.mongodb.DBCollection;
            import  java.io.*;
            import java.util.*;
            import com.mongodb.*;
            public class MongoDemo {
public static void main(String[] args){
try{
```

```
//---------- Connecting DataBase -------------------------//
MongoClient mongoClient = new MongoClient( "localhost" , 27017 );
```

```
//---------- Creating DataBase ---------------------------//
MongoDatabase db = mongoClient.getDatabase("kls");
//---------- Creating Collection -------------------------//
MongoCollection<Document> table = db.getCollection("employee");
//---------- Creating Document ---------------------------//
            Document filter = new Document("_id",new
ObjectId(id);
            Bson newValue = new Document("email", "shbcjhvs");
            Bson updateOperationDocument = new Document("$set",
newValue);
            table.updateOne(filter, updateOperationDocument);
}}}
```

Story 2 – Project Work

About Project

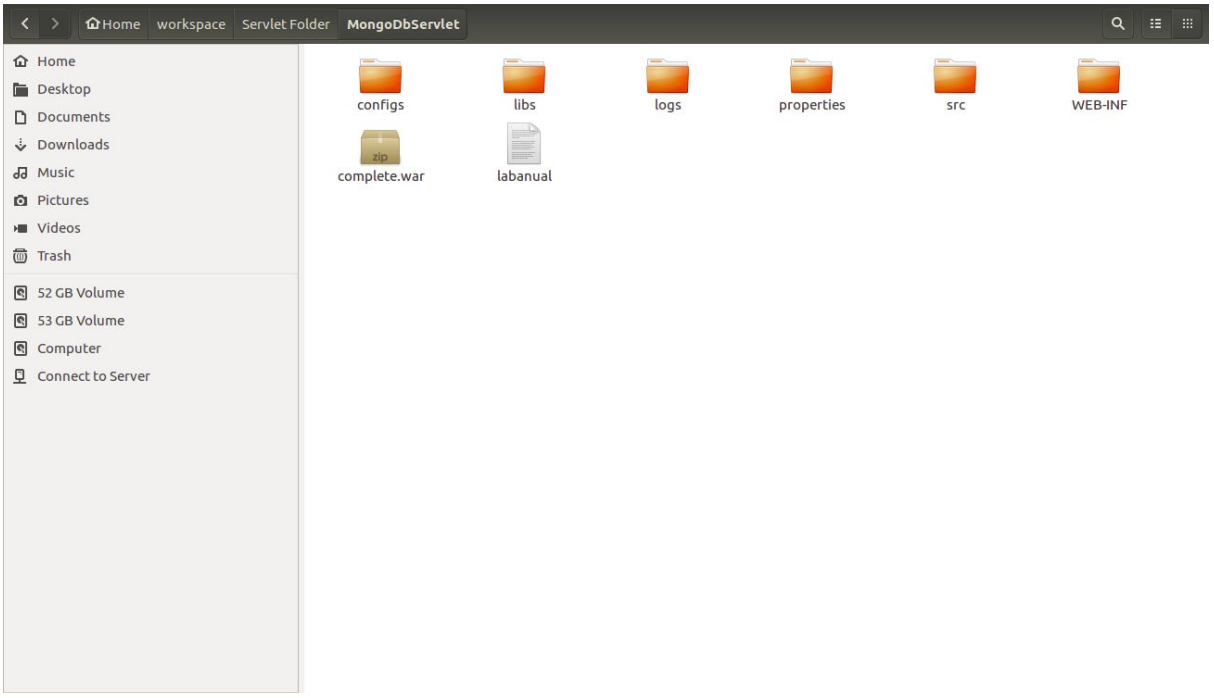
#In this project first we create a login form in which user enter userid and pass if userId and pass correct we redirect this page to login page if userid and password inocorrect it will redirect to error page

About Login Page : In this we show all data present into database here we give the functionality to user to add user remove user update user records here also we give a logout button which redirect us to homepage.

About Add Page : In this page we take data from user and insert into database

About Update Page : We update user record here user click on any employee with that employee we pass that employee id and on this page we update that employee record based on id

Task 1- Create Directory Structure



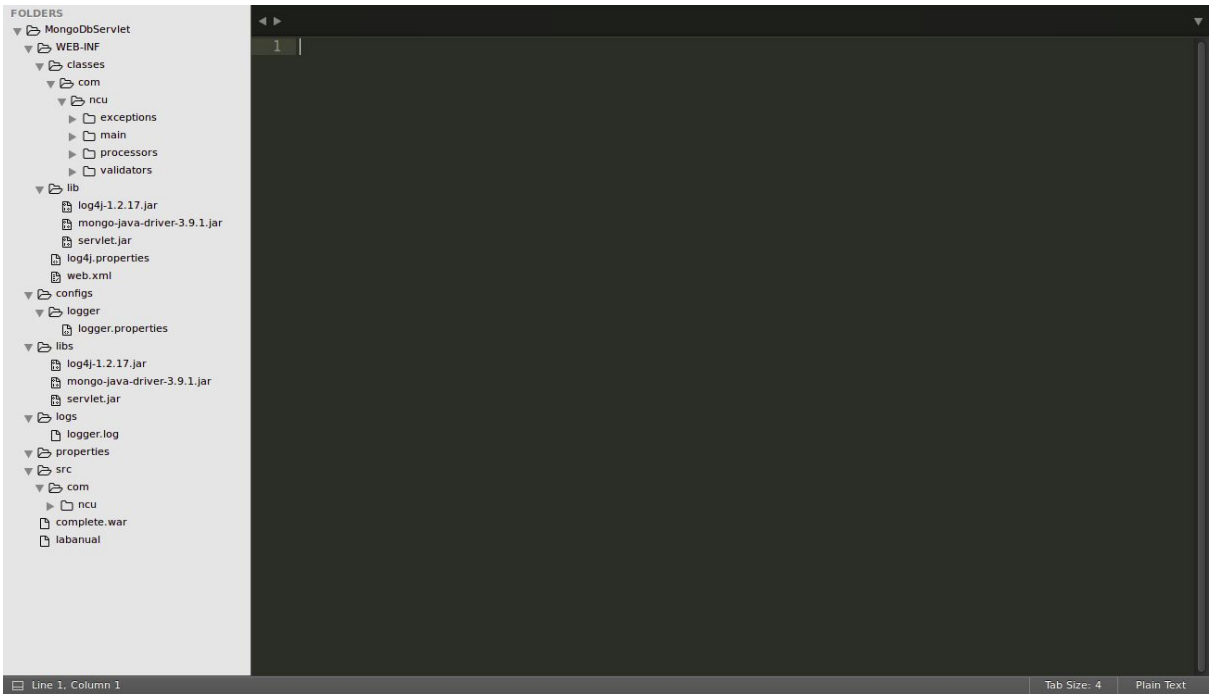
Src folder: In this folder we put our .java files

WEB-INF : In this folder we put out web.xml and inside this another folder classes in which we put .class file

libs folder : In this folder we put our jar files

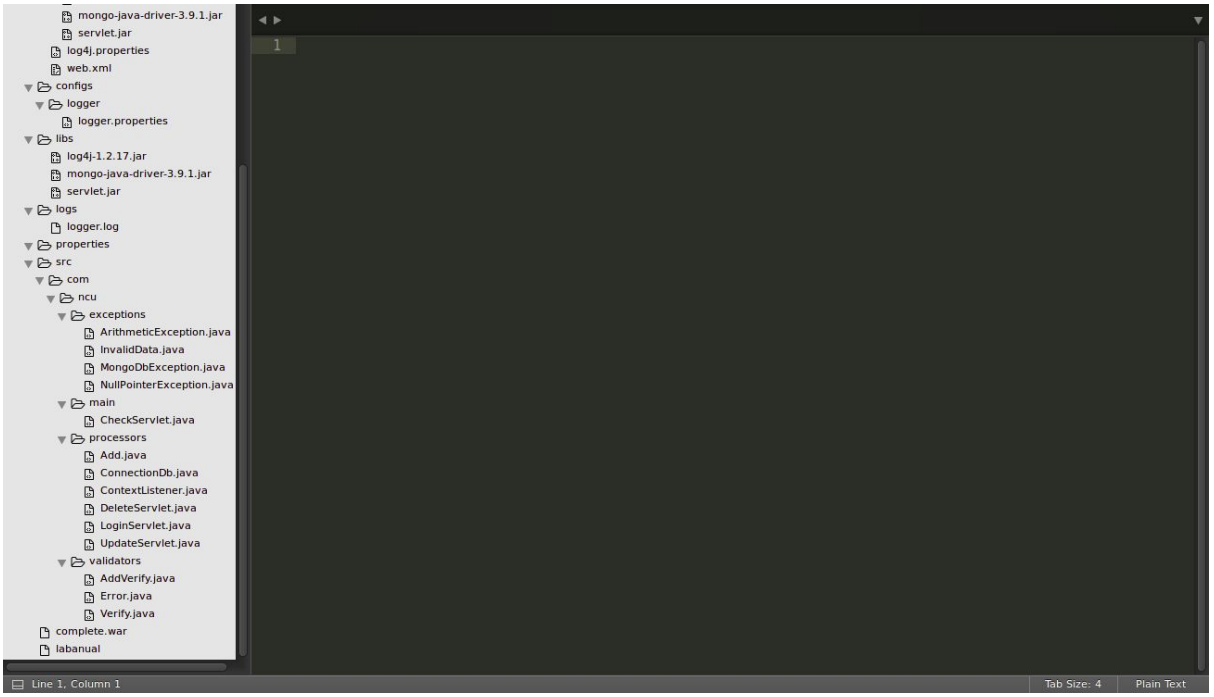
logs folder: In this we put our logger file where logger data come

properties folder : In this we put our .properties file



Directory structure of complete project

Task 2: Make .java file according to this directory structure



Task 3: Make Add.java file in com.ncu.processors package

```
package com.ncu.processors;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;

import java.io.IOException;
import java.io.PrintWriter;

import com.mongodb.DB;
import com.mongodb.DBObject;
import com.mongodb.BasicDBObject;
import com.mongodb.DBCollection;
import com.mongodb.MongoClient;
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoDatabase;
import org.bson.types.ObjectId;
public class Add extends HttpServlet{
public void doGet(HttpServletRequest req,HttpServletResponse res)
throws ServletException,IOException
{


String getValue=req.getParameter("uname");
if(!getValue.equals(null))
{

String str=req.getParameter("uname");
String str2=req.getParameter("mob");
String str1=req.getParameter("pass");
String str3=req.getParameter("email");
try
{
      MongoClient mongoClient = new MongoClient( "localhost" ,
27017 );
      DB db = mongoClient.getDB("at");
            DBCollection collection = db.getCollection("people");
              DBObject document1 = new BasicDBObject();
```

```
document1.put("name", str);
document1.put("pass", str1);
document1.put("mobile Number",str2);
document1.put("email",str3);

collection.insert(document1);
res.sendRedirect("login");
}
catch(Exception e)
{

}
}
out.println("hello");
 out.println("<input type='submit' value='Submit'></form>");

}
}
```

Task 4: create DeleteServlet.java file in com.ncu.processors

```
package com.ncu.processors;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;

import java.io.PrintWriter;
import java.io.IOException;

import org.bson.types.ObjectId;
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoDatabase;
import com.mongodb.DBCollection;
```

```
import com.mongodb.MongoClient;
import org.bson.Document;
import com.mongodb.client.MongoCollection;
public class DeleteServlet extends HttpServlet{
public void doGet(HttpServletRequest req,HttpServletResponse res)
throws ServletException,IOException
{
res.setContentType("text/html");
PrintWriter out=res.getWriter();
String str=req.getParameter("uname");

out.println(str);
try
{

  MongoClient mongoClient = new MongoClient( "localhost" , 27017
);
          MongoDatabase db = mongoClient.getDatabase("at");
MongoCollection<Document> collection = db.getCollection("people");
     collection.deleteOne(new Document("_id", new ObjectId(str)));
     out.println("Record Deleted");

    res.sendRedirect("login");


      }
      catch(Exception e)
      {
     out.println("Exception");
      }
}
}
```

Task 5: Create LoginServlet.java file com.ncu.processors

```
package com.ncu.processors;
```

```
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;

import java.io.IOException;
import java.io.PrintWriter;

import com.mongodb.DB;
import com.mongodb.BasicDBObject;
import com.mongodb.DBCollection;
import com.mongodb.MongoClient;
import com.mongodb.DBCursor;
import com.mongodb.DBObject;
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoDatabase;
import org.bson.types.ObjectId;
import java.io.File;

import org.apache.log4j.Logger;

public class LoginServlet extends HttpServlet{
  public void doGet(HttpServletRequest req,HttpServletResponse
res)
throws ServletException,IOException
{
res.setContentType("text/html");
PrintWriter out=res.getWriter();


try
{
  MongoClient mongoClient = new MongoClient( "localhost" , 27017
);
  DB db = mongoClient.getDB("at");
            DBCollection collection = db.getCollection("people");
            DBCursor dbo = collection.find();
```

```
            out.println("<html><body style='background-color:
darkorange';> ");
        out.println("<head><h1 align='center'><u>User
Management</u></h1></head>");
 out.print("<table width=80% border=1>");
 out.println("<form action='add'><input type='submit' value='Add
New'></form>");
 out.print("<caption><h3>User Data:</h3></caption>");
 out.println("<form action='check'><input type='submit'
value='Logout'></form>");

  out.println("<th>Name</th>");
  out.println("<th>Password</th>");
     out.println("<th>Email</th>");
     out.println("<th>Mobile</th>");
     out.println("<th>ID</th>");
     out.println("<th>Options</th>");
          out.println("</tr>");
          LOGGER.info("info");
 while(dbo.hasNext())
    {
       DBObject dbq=dbo.next();
       ObjectId id = (ObjectId) dbq.get( "_id" );
          String name=(String) dbq.get("name");
      String pass=(String) dbq.get("pass");
      String mob=(String) dbq.get("mobile Number");
      String email=(String) dbq.get("email");
       out.println("<tr>");
        out.println("<td>"+name+"</td>");
     out.println("<td>"+pass+"</td>");
         out.println("<td>"+email+"</td>");
     out.println("<td>"+mob+"</td>");
     out.println("<td>"+id+"</td>");
     out.println("<td><a
href='delete?uname="+id+"'>Delete</a></td> ");
      out.println("<td><a
href='update?uname="+id+"'>Edit</a></td></form>");

  }

}
```

```
catch(Exception e)
{

}}}
```

Task 6 : Create UpdateServlet.java file in com.ncu.processors

```
 package com.ncu.processors;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServlet;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServletResponse;

 import java.io.IOException;
import java.io.PrintWriter;

import com.mongodb.BasicDBObject;
import com.mongodb.DB;
import com.mongodb.DBCollection;
import com.mongodb.*;
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoDatabase;
import com.mongodb.MongoClient;
import org.bson.*;
import org.bson.types.ObjectId;
import org.bson.Document;
import org.bson.conversions.Bson;
public class UpdateServlet extends HttpServlet
{
```

```
     public void doGet(HttpServletRequest
req,HttpServletResponse res)
     throws ServletException,IOException
     {
          res.setContentType("text/html");
          PrintWriter out=res.getWriter();
          String id=req.getParameter("uname");

          out.println("<form  action='update'
method='GET'>");
          Number", mob);
               infoObject.put("pass",pass);
```

```
infoObject.put("email",email);
                Bson updateObject = new Document("$set",
infoObject);

                collection.updateOne(Query, updateObject);
                res.sendRedirect("login");out.println("<html><body
style='background-color: darkorange;height:200px';>");
          out.println("<head><h1 align='center'><u>Update User
Record</u></h1></head>");
          out.print("<table width=30% align='center' border=1>");
          out.println("<tr><td>Update Mobile</td><td><input
type='text' name='mob'></td>");
          out.println("<tr><td>Update Password</td><td><input
type='text' name='pass'></td>");
        out.println("<tr><td> id</td><td><input type='hidden'
type='text' name='getId' value='"+id+"'></td>");
          out.println("<tr><td>Update Email</td><td><input
type='text' name='email'></td></tr>");
          out.println("<tr><td><input type='submit' name='submit'
value='submit'><td></tr></form>");
          String mob=req.getParameter("mob");
          String pass=req.getParameter("pass");
          String email=req.getParameter("email");
          String newId=req.getParameter("getId");
```

```
            //out.println("New ID"+newId);
        if(!newId.equals(null)){
            try
            {
                MongoClient mongoClient = new MongoClient(
"localhost" , 27017 );
                MongoDatabase db = mongoClient.getDatabase("at");
                MongoCollection<Document> collection =
db.getCollection("people");


                Document Query = new Document("_id", new
ObjectId(newId));

                // Creating Object to update value .
                Document infoObject = new Document("mobile
            }
            catch(Exception e)
            {
      out.println("404");
            }
        }
        else
        {
            out.println("else");
        }
      }
}
```

Task 7 : Create a Verify.java file in com.ncu.validators package

```
package com.ncu.validators;
import com.ncu.exceptions.*;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.ServletException;
```

```
import com.ncu.exceptions.*;
import java.io.IOException;
import java.io.PrintWriter;

import java.io.IOException;
public class AddVerify extends HttpServlet
{

      public void doGet(HttpServletRequest req,HttpServletResponse
res)
throws ServletException,IOException
{
      res.setContentType("text/html");
PrintWriter out=res.getWriter();
out.println("<form  action='add' method='GET'>");
out.println("<html><body style='background-color: darkorange';>
");
out.println("<head><h1 align='center' style='margin-top:30px;'
><u>Add User</u></h1></head>");
out.print("<table align='center'
style='margin-top:70px;height:300px;' border=1>");
out.println("<tr><td>Name</td><td><input type='text'
name='uname'></td></tr>");
out.println("<tr><td>Mobile No</td><td><input type='text'
name='mob' ></td></tr>");
out.println("<tr><td>Password</td><td><input type='password'
name='pass'></td></tr>");
out.println("<tr><td>Email</td><td><input type='text'
name='email'></td></tr></table>");
 out.println("<input type='submit' align='center'
style='margin-left:630;width:90px;height:40px;margin-top:30px;'
value='submit'>");

String str=req.getParameter("uname");
String str2=req.getParameter("mob");
String str1=req.getParameter("pass");
String str3=req.getParameter("email");


}
else
```

```
{     res.sendRedirect("add");}}}
```

Task 8 - Create Error.java file in com.ncu.validators package

```
package com.ncu.validators;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.ServletException;

import java.io.IOException;
import java.io.PrintWriter;

public class Error extends HttpServlet{
public void doGet(HttpServletRequest req,HttpServletResponse res)
throws ServletException,IOException
{
res.setContentType("text/html");
PrintWriter out=res.getWriter();
out.println("<html><body style='background-color: darkorange';>");
out.println("<head><h1 align='center'
style='margin-top:30px;'><u>UserName And Password
Incorrect</u></h1></head>");

}
}
```

First set classpath of Servlet And MongoDb jar file

Task 9: After making these java file according to this directory structure first go to Main package open terminal and run this command

```
  javac -d
"/home/kls105/workspace/ServletFolder/MongoDbServlet/WEB-INF/class
es" *.java
```

This command create a.class file on this location

Task 10 : After That go to the processors package and run same command

```
  javac -d
"/home/kls105/workspace/ServletFolder/MongoDbServlet/WEB-INF/class
es" *.java
```

Task 11 : After that go to the validators package and run this command

```
  javac -d
"/home/kls105/workspace/ServletFolder/MongoDbServlet/WEB-INF/class
es" *.java
```

Task 12 : After that go to root directory open terminal and make war file

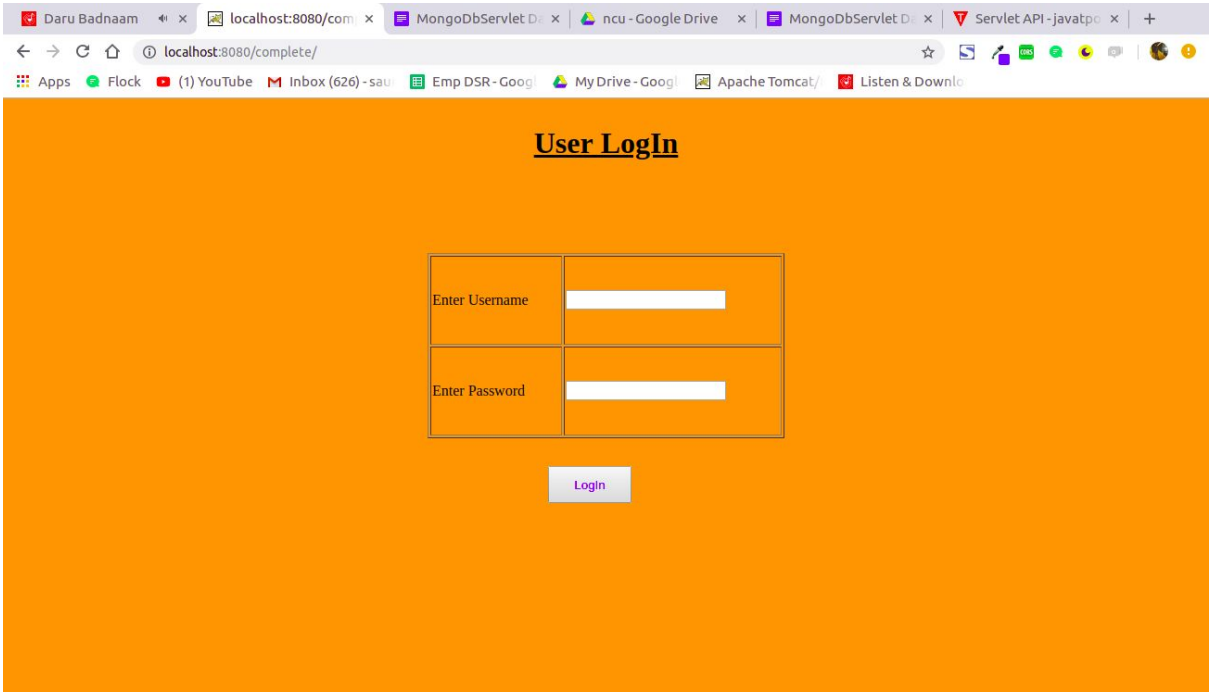
```
jar cvf mongo.war *
```

Task 13 : Start tomcat server

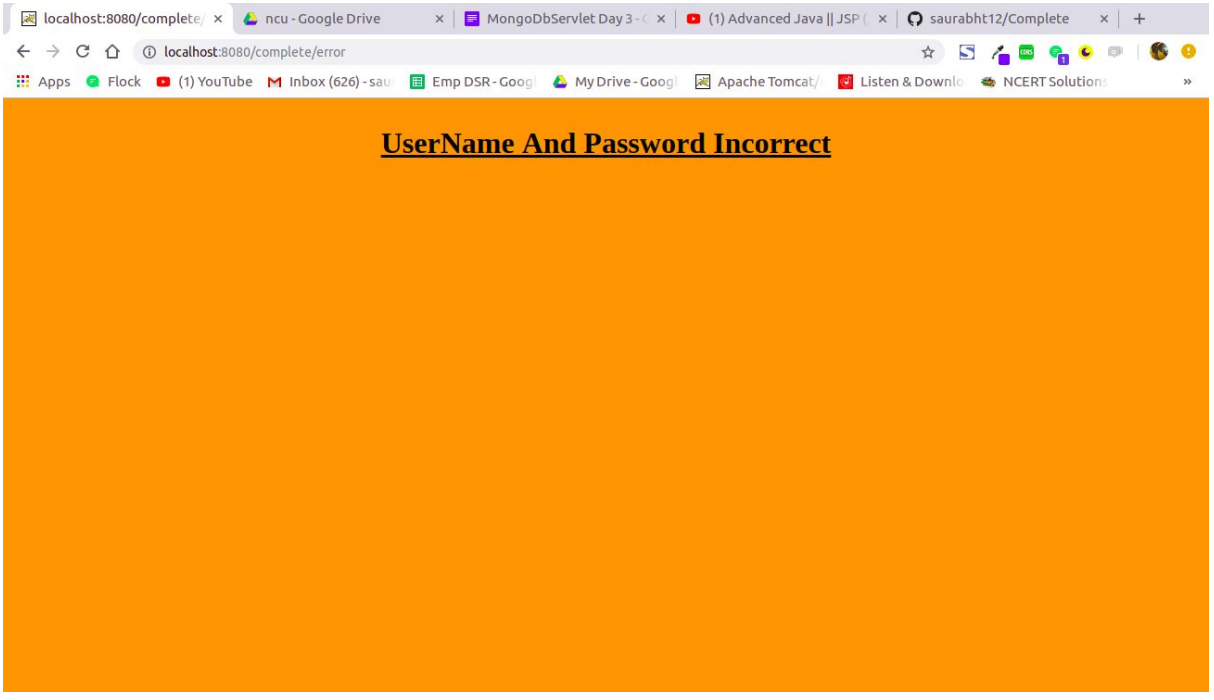
Task 14 : Open browser and type this url
http://localhost:8080

Task 15 : deploy your war file and after succesfully deploy click on war file to run project

Task 16: After clicking on war file first page be like



Task 17 : This your login page here you will provide data which verify from database if data given by you correct login page open otherwise error page open



Task 18: Login page if you enter the correct data here Add New to add new record to Database and logout button to redirect you to homepage