

CURSO : LENGUAJE DE
PROGRAMACIÓN II (1894)

PROFESOR : RAEL MOSTACERO MENDOZA
SEMESTRE : 2023-II

CICLO : IV

SECCIÓN : T4KN

GRUPO : 01

FECHA : JUEVES 26 DE OCTUBRE DE 2023

DURACIÓN : 90 MINUTOS

NOTA

ALUMNO (A) : David Aaron Sanchez

EVALUACIÓN DE LABORATORIO 2

Consideraciones generales:

- Se considera el orden del código y cumplir con los estándares de programación JAVA(MVC), así como también el uso de Spring boot.
- Está permitido el uso o consulta de cuadernos, separatas, libros o cualquier material de la asignatura durante el desarrollo de la evaluación.
- La evaluación debe tener la siguiente estructura LP2-T4KN-Nombre Apellido.
- Se bonificará con un punto si el código esta simplificado.

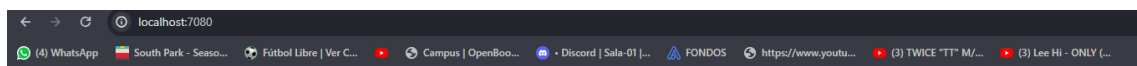
SPRING BASICO

- El examen iniciará a las 16:00 y se entregará a las 23:59 y luego se comprimirá y se subirá a la plataforma de blackboard.
- El examen se descargará de la plataforma, este examen se encuentra en formato de Word, para que puedas agregar la captura de pantalla por cada pregunta.
- La captura de pantalla involucra la solución de cada pregunta, esta captura debe contener además el URL, donde se encuentra la ruta y el puerto y la pagina web del navegador, además debes pegarla en el examen
- Luego debe juntar el archivo de Word y la o las aplicaciones de eclipse, para comprimirlo con el nombre LP2-T4KN-Nombre Apellido.
- Finalmente debes subirlo en la plataforma de Blackboard

1. Desarrollar una aplicación con el nombre **pspring1**, que permita enviar el **mensaje** ("Examen de Laboratorio 2 - tu nombre"), hacia el navegador de internet, usando el **puerto 7080**, sin utilizar ninguna página web.

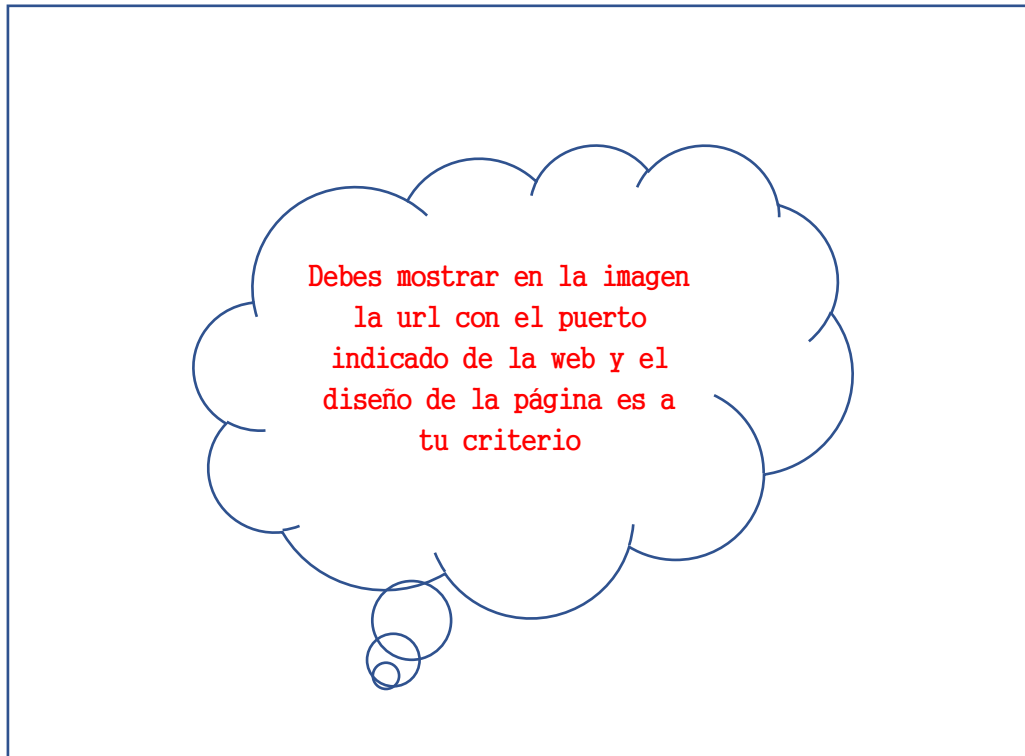
Debes mostrar en la imagen
la url con el puerto
indicado de la web y el
mensaje e indicando tu
nombre

```
Primera.java x
Pregunta1-Spring > sanchez > src > main > java > com > david > sanchez > controller > Primera.java > ...
1 package com.david.sanchez.controller;
2
3 import org.springframework.web.bind.annotation.GetMapping;
4 import org.springframework.web.bind.annotation.RestController;
5
6 @RestController
7 {
8     public class Primera {
9         @GetMapping(value = "/")
10        public String saludo() {
11            return "Examen de Laboratorio 2 - David Aaron Sanchez";
12        }
13    }
14 }
15
```



Examen de Laboratorio 2 - David Aaron Sanchez

2. Desarrollar una aplicación con el nombre **pspring2**, deberás diseñar una página web y mostrarla usando el **puerto 8080** y la **ruta: /**



```
package com.david.segunda.controller;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;

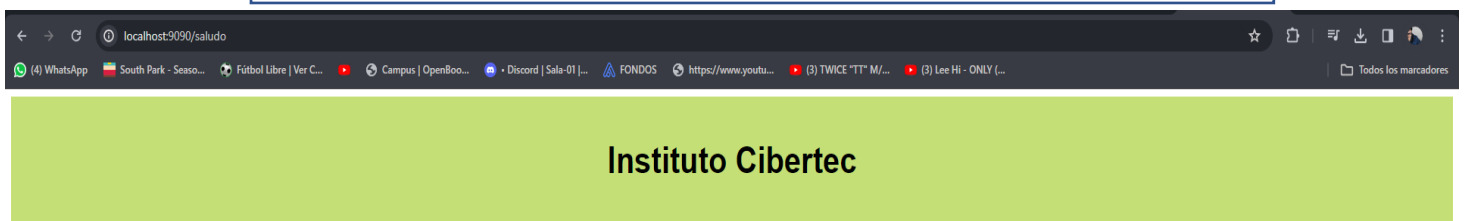
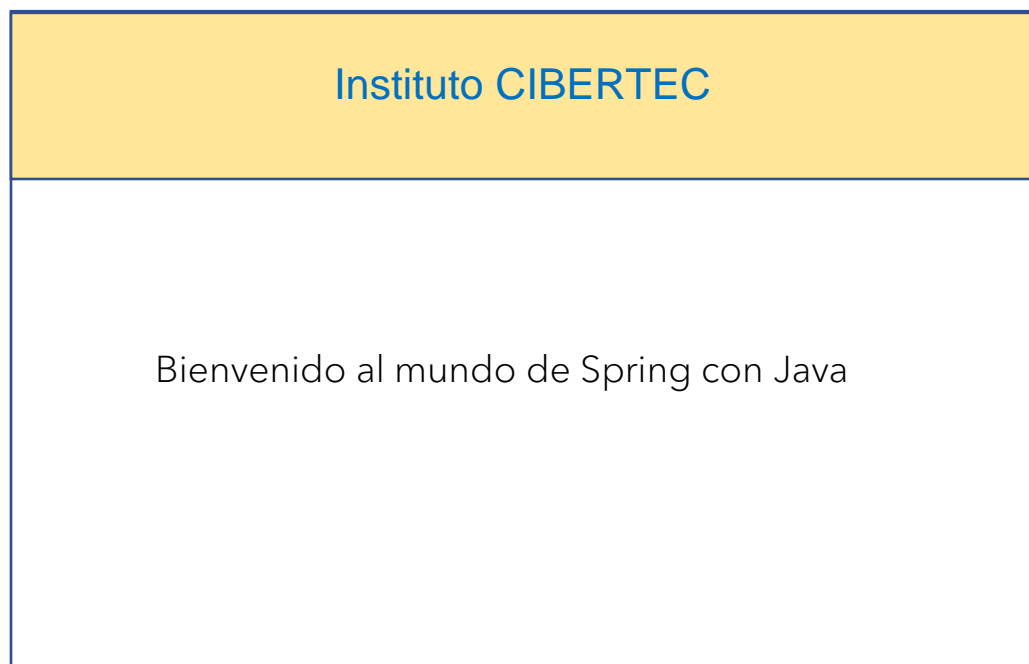
@Controller
public class Segunda {
    @GetMapping(value = "/")
    public String home() {
        return "index";
    }
}
```

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <style>...
7 </style>
8 <title>Inicio</title>
9 </head>
10 <body> <!--No se porque no visualiza en la imagen-->
11   <header>
12     <h1 class="titulo">David Sanchez
13     <span>Full Stack</span> </h1> <!--No es codigo reciclado este es mi proyecto q he estado avanzando-->
14   </header>
15   <div class="nav-bg">
16     <nav class="navegacion-principal contenedor">
17       <a href="#">Inicio</a>
18       <a href="#">Notas</a>
19       <a href="#">Sobre mi</a>
20       <a href="#">Contacto</a>
21     </nav>
22   </div>
23
24   <section class="hero">
25     <div class="contenido-hero">
26       <h2>Desarrollo web en PHP
27       <br>
28       <span> Conocimientos Amplios Sistemas de Ventas
29       <br>
30       y Gestores de Archivos</span></h2>
31       <div class="ubicacion">
32         <p>Lima, Peru</p>
33       </div>
34       <a class="boton" href="#">Contactar</a>
35     </div> <!--.contenido-hero-->
36   </section>
37
38 </body>
39 </html>

```

3. Desarrollar una aplicación con el nombre **pspring3** que permita enviar un **saludo** ("Bienvenido al mundo de Spring con Java") desde el **controlador** hacia la **página web** con el siguiente diseño propuesto, utilizando **variables** y **thymeleaf**, y usando la URL con el **puerto 9090** y la **ruta: /saludo**



Bienvenido al mundo de Spring con Java

```
tercera.java  instituto.html x
tercera > src > main > resources > templates > instituto.html > body
1  <!DOCTYPE html>
2  <head>
3    <title>Instituto CIBERTEC</title>
4  >
5    <style>...
58  </style>
59  </head>
60  <body>
61    <header>
62      <h1>Instituto Cibertec</h1>
63    </header>
64    <section style="text-align: center;">
65      <p th:text="${saludo}"></p>
66    </section>
```

```
tercera.java x  instituto.html
tercera > src > main > java > com > david > tercera > controller > tercera.java > ...
1  package com.david.tercera.controller;
2
3  import org.springframework.ui.Model;
4  import org.springframework.web.bind.annotation.GetMapping;
5  import org.springframework.stereotype.Controller;
6
7  @Controller
8  public class tercera {
9
10     @GetMapping("/saludo")
11     public String mostrarMensaje(Model model) {
```

4. Desarrollar una aplicación con el nombre **pspring4** que permita mostrar los **datos de un producto** (**código, nombre, precio, cantidad**) desde el **controlador** hacia la **página web** con el siguiente **diseño propuesto**, utilizando **variables, thymeleaf** y una **clase producto**. Y usando la URL con el **puerto 7050** y la **ruta: /producto/único**

Detalle del Producto

Código:

P001

Nombre:

Arroz

Precio:

150.00

Cantidad:

1500

```
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.GetMapping;

import com.david.tercera.modelo.Producto;

@Controller
public class Codigo {
    @GetMapping("/producto/unico")
    public String mostrar(Model model) {
        Producto producto = new Producto();
        producto.setCodigo(codigo:"P001");
        producto.setNombre(nombre:"Arroz");
        producto.setPrecio(precio:150);
        producto.setCantidad(cantidad:1500);
        model.addAttribute(attributeName:"producto", producto);
        return "producto";
    }
}
```

```

public class Producto {
    private String codigo;
    private String nombre;
    private double precio;
    private int cantidad;

    public String getCodigo() {
        return codigo;
    }

    public void setCodigo(String codigo) {
        this.codigo = codigo;
    }

    public String getNombre() {
        return nombre;
    }

    public void setNombre(String nombre) {
        this.nombre = nombre;
    }

    public double getPrecio() {
        return precio;
    }

    public void setPrecio(double precio) {
        this.precio = precio;
    }

    public int getCantidad() {
        return cantidad;
    }

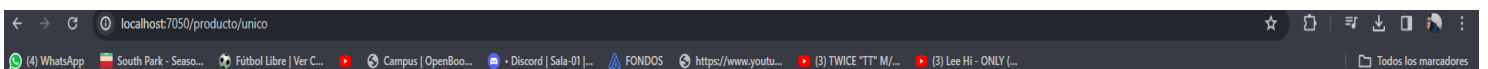
    public void setCantidad(int cantidad) {
        this.cantidad = cantidad;
    }
}

```

```

<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
    <meta charset="UTF-8">
    <title>Maqueta de producto</title>
</head>
<style>...
</style>
<body>
    <header>
        <h1>Detalle del Producto</h1>
    </header>
    <section>
        <ul>
            <p>Código: <span th:text="${producto.codigo}"></span></p>
            <p>Nombre: <span th:text="${producto.nombre}"></span></p>
            <p>Precio: <span th:text="${producto.precio}"></span></p>
            <p>Cantidad: <span th:text="${producto.cantidad}"></span></p>
        </ul>
    </section>
</body>
</html>

```



Detalle del Producto

Código: P001

Nombre: Arroz

Precio: 150.0

Cantidad: 1500

5. Desarrollar una aplicación con el nombre **pspring5** que permita mostrar tres estudiantes (**código, nombre, curso, examen final**) desde el **controlador** hacia la **página web** con el siguiente diseño propuesto, utilizando **variables, thymeleaf, tabla, Bootstrap** y una **clase producto**. Y usando la URL con el **puerto 7080** y la **ruta: /estudiantes/lista**

Listado de Estudiantes			
Código	Nombre	Curso	ExaParcial
Alum001	Carlos	LP II	18
Alum002	Karla	LP I	15
Alum003	Miguel	POO	20

```
public class Estudiantes {
    private String codigo;
    private String nombre;
    private String curso;
    private double examenFinal;

    public Estudiantes() {
    }

    public Estudiantes(String codigo, String nombre, String curso, double examenFinal) {
        this.codigo = codigo;
        this.nombre = nombre;
        this.curso = curso;
        this.examenFinal = examenFinal;
    }

    public String getCodigo() {
        return codigo;
    }

    public void setCodigo(String codigo) {
        this.codigo = codigo;
    }

    public String getNombre() {
        return nombre;
    }

    public void setNombre(String nombre) {
        this.nombre = nombre;
    }

    public String getCurso() {
        return curso;
    }

    public void setCurso(String curso) {
        this.curso = curso;
    }

    public double getExamenFinal() {
        return examenFinal;
    }
}
```

```
package com.david.quinta.controller;

import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.GetMapping;

import java.util.ArrayList;
import java.util.List;

import com.david.quinta.modelo.Estudiantes;

@Controller
public class Estudiantee {

    @GetMapping("/estudiantes/lista")
    public String mostrarEstudiantes(Model model) {
        List<Estudiantes> estudiantes = new ArrayList<>();
        estudiantes.add(new Estudiantes(codigo:"Alum001", nombre:"Carlos", curso:"LP2", examenFinal:18));
        estudiantes.add(new Estudiantes(codigo:"Alum002", nombre:"Karla", curso:"LP1", examenFinal:15));
        estudiantes.add(new Estudiantes(codigo:"Alum093", nombre:"Miguel", curso:"LP2", examenFinal:20));

        model.addAttribute(attributeName:"estudiantes", estudiantes);

        return "estudiantes";
    }
}
```



```

<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
  <title>Lista de Estudiantes</title>
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css">
</head>
<body>
  <header>
    <h1>Lista de Estudiantes</h1>
  </header>
  <section>
    <table class="table table-striped" >
      <thead>
        <tr>
          <th>Código</th>
          <th>Nombre</th>
          <th>Curso</th>
          <th>Examen Final</th>
          <th></th>
        </tr>
      </thead>
      <tbody>
        <tr th:each="estudiante : ${estudiantes}">
          <td th:text="${estudiante.codigo}"></td>
          <td th:text="${estudiante.nombre}"></td>
          <td th:text="${estudiante.curso}"></td>
          <td th:text="${estudiante.examenFinal}"></td>
          <td>
            <i class="fas fa-trash-alt" style="color: red;"></i>
          </td>
        </tr>
      </tbody>
    </table>
  </section>
</body>
</html>

```

localhost:7080/estudiantes/lista

South Park - Seaso... Fútbol Libre | Ver C... Campus | OpenBoo... Discord | Sala-01 |... FONDOS https://www.youtu... (3) TWICE "TT" M/... (3) Lee Hi - ONLY (... Todos los

Lista de Estudiantes

Código	Nombre	Curso	Examen Final	
Alum001	Carlos	LP2	18.0	<i>fas fa-trash-alt</i>
Alum002	Karla	LP1	15.0	<i>fas fa-trash-alt</i>
Alum093	Miguel	LP2	20.0	<i>fas fa-trash-alt</i>

6. Desarrollar una aplicación con el nombre **pspring6** que permita mostrar un formulario de contacto, desde el **controlador** hacia la **página web** con el siguiente **diseño** **propuesto**, utilizando **variables**, **thymeleaf**, **Bootstrap** y una **clase producto**. Y usando la URL con el **puerto 9090** y la **ruta: /cliente/contacto**

Formulario de Contacto

Código:

Cliente:

Teléfono:

Distrito:

Edad:

Guardar

```
1 package com.david.ultima.controller;
2 import org.springframework.stereotype.Controller;
3 import org.springframework.web.bind.annotation.GetMapping;
4 import org.springframework.ui.Model;
5 import com.david.ultima.modelo.Regis;
6
7 @Controller
8 public class Registro {
9
10     @GetMapping("/cliente/contacto")
11     public String mostrarFormulario(Model model) {
12         Regis regis = new Regis();
13         regis.setCodigo(codigo:"01");
14         regis.setNombre(nombre:"David");
15         regis.setTelefono(telefono:"984290861");
16         regis.setDistrito(distrito:"Comas");
17         regis.setEdad(edad:19);
18
19         model.addAttribute(attributeName:"regis", regis);
20
21         return "formulario";
22     }
23 }
```

```
private String codigo;
private String nombre;
private String telefono;
private String distrito;
private int edad;

public Regis() {
}

public Regis(String codigo, String nombre, String telefono, String distrito, int edad) {
    this.codigo = codigo;
    this.nombre = nombre;
    this.telefono = telefono;
    this.distrito = distrito;
    this.edad = edad;
}

public String getCodigo() {
    return codigo;
}

public void setCodigo(String codigo) {
    this.codigo = codigo;
}

public String getNombre() {
    return nombre;
}

public void setNombre(String nombre) {
    this.nombre = nombre;
}

public String getTelefono() {
    return telefono;
}

public void setTelefono(String telefono) {
    this.telefono = telefono;
}

public String getDistrito() {
}
```

```

1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3 <head>
4   <title>Formulario</title>
5   <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
6 </head>
7 <body>
8   <div class="container">
9     <header class="bg-success p-3 text-center text-light mb-5">
10       <h1>Formulario de Contacto</h1>
11     </header>
12     <main class="container">
13       <form th:object="${regis}" th:action="@{/guardar}" method="post">
14         <div class="form-group">
15           <label for="codigo" class="col-sm-2 col-form-label">Código</label>
16           <div class="col-sm-10">
17             <input type="text" class="form-control" id="codigo" th:field="${codigo}">
18           </div>
19         </div>
20         <div class="form-group">
21           <label for="nombre" class="col-sm-2 col-form-label">Cliente</label>
22           <div class="col-sm-10">
23             <input type="text" class="form-control" id="nombre" th:field="${nombre}">
24           </div>
25         </div>
26         <div class="form-group">
27           <label for="telefono" class="col-sm-2 col-form-label">Teléfono</label>
28           <div class="col-sm-10">
29             <input type="text" class="form-control" id="telefono" th:field="${telefono}">
30           </div>
31         </div>
32         <div class="form-group">
33           <label for="distrito" class="col-sm-2 col-form-label">Distrito</label>
34           <div class="col-sm-10">
35             <input type="text" class="form-control" id="distrito" th:field="${distrito}">
36           </div>
37         </div>
38         <div class="form-group">
39           <label for="edad" class="col-sm-2 col-form-label">Edad</label>
40           <div class="col-sm-10">
41             <input type="number" class="form-control" id="edad" th:field="${edad}">
42           </div>
43         </div>
44         <button type="submit" class="btn btn-primary">Guardar</button>
45       </form>
46     </main>
47   </div>
48 </body>
49 </html>

```

Formulario de Contacto

Código

01

Cliente

David

Teléfono

984290861

Distrito

Comas

Edad

19

Rubrica:

Preguntas	Completo		Avance		Nada o Casi nada	
1	2	Mostrar mensaje en el navegador y el puerto de salida	1	Mostrar mensaje en el navegador con otro puerto	0	No mostrar nada
2	2.5	Mostrar página web, la ruta de acceso y puerto de salida	2	Mostrar página web incompleta, pero la ruta de acceso y puerto de salida completa	0	No mostrar nada
3	2.5	Mostrar página web, mensaje, ruta de acceso y puerto de salida	2	Mostrar página web sin el mensaje, pero la ruta de acceso y puerto de salida completa	1	No mostrar nada, excepto la ruta y puerto
4	3	Mostrar página web, registro del producto, ruta de acceso y puerto de salida	2	Mostrar página web, sin el registro del producto, pero la ruta de acceso y puerto de salida completa	1	No mostrar nada, excepto la ruta y puerto
5	5	Mostrar página web, Lista de estudiantes, la tabla, ruta de acceso y puerto de salida	4	Mostrar página web, sin Lista de estudiantes, la tabla, pero con ruta de acceso y puerto de salida completa	2	No mostrar nada, excepto la ruta y puerto
6	5	Mostrar página web, formulario de ingreso, botón, ruta de acceso y puerto de salida.	4	Mostrar página web, sin formulario de ingreso, botón, pweo con la ruta de acceso y puerto de salida completa	2	No mostrar nada, excepto la ruta y puerto