

Alumnos: Agraso Joaquin

Arrua Fiama

Flores Sergio

**Carrera:** Tecnicatura Universitaria en Programación

**Año:** 1º año.

**Materias:** Laboratorio de computación 2 y Programación 2.

**Profesores:** García Gustavo - Guzmán Celina.

NOMBRE DEL PROYECTO:

**HOTEL**

¿QUÉ HACE?

Muestra por pantalla en el browser una tabla con las reservas de un hotel, si está o no reservada, capacidad, precio y numero de habitación.

PLAN DE DESARROLLO:

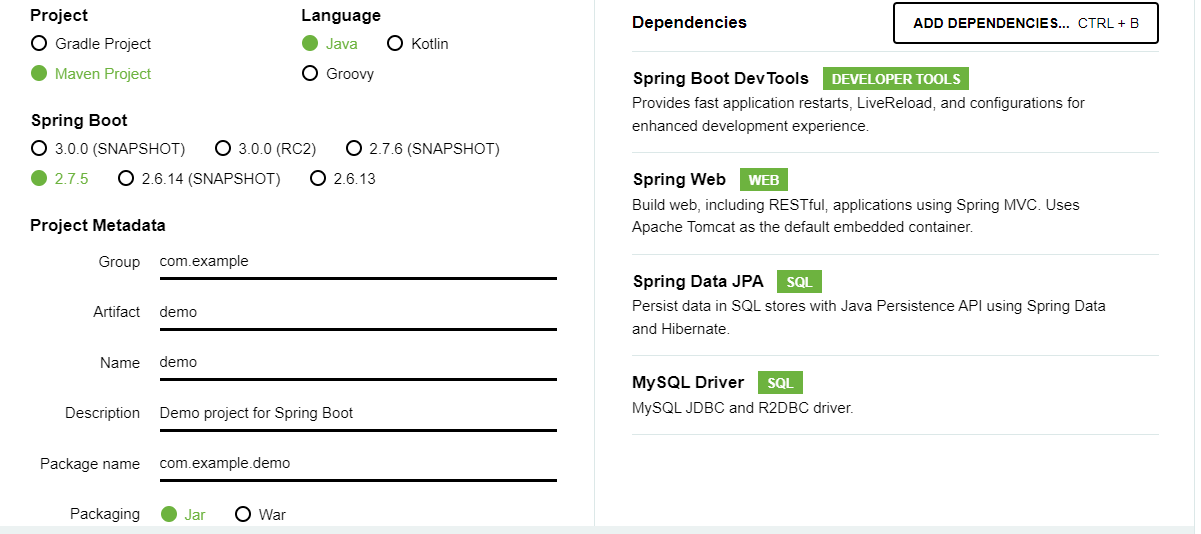
**ESPECIFICACIONES:**

* Servidor backend armado en Springboot con Maven.
* Java 17.
* Base de datos SQL.
* Debe tener controlador REST.
* Controlador debe ser capaz de responder a requests de GET y POST.
* Sin frontend, las pruebas se realizan en postman.

Se crea el proyecto, directamente desde Visual Studio Code o desde <https://start.spring.io/> en este caso usamos esta última opción.

Se generó un proyecto:

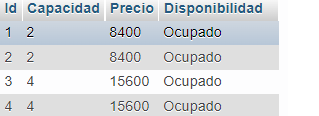
* Spring Boot 2.7.5
* Maven Project
* En lenguaje Java con package Java 17.
* Y se seleccionaron las dependencias SpringBoot Dev Tools – Spring Web – Spring JPA yMySQL Driver.



Iniciamos el proyecto creando las tablas en PhpMyAdmi. La base de datos se llama **Hotel** con la clase **Habitaciones.**

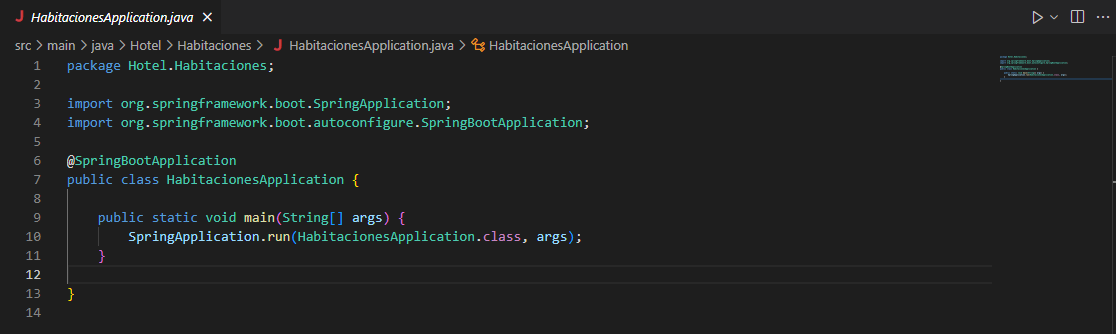
Usuario: root

Contraseña:

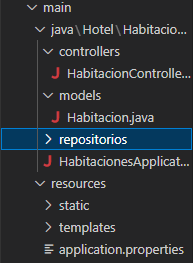


Creamos las diferentes carpetas y clases para cada una de ellas.

Una vez creado se guarda el proyecto y se abre en Visual Studio Code o el entorno de desarrollo deseado. Al abrirlo encontraremos solo la clase principal en este caso la clase Habitaciones:



Se crean los demás paquetes que serán: controllers, models y repositorios:



**Java\PROYECTO\Habitaciones\models\Habitacion.java**

@Entity

public class Habitacion {

    @Id

    @GeneratedValue(strategy = GenerationType.IDENTITY)

    private Long id;

    private int Capacidad;

    private int Precio;

    private String Disponibilidad;

Indicamos los nombres de la varibles.

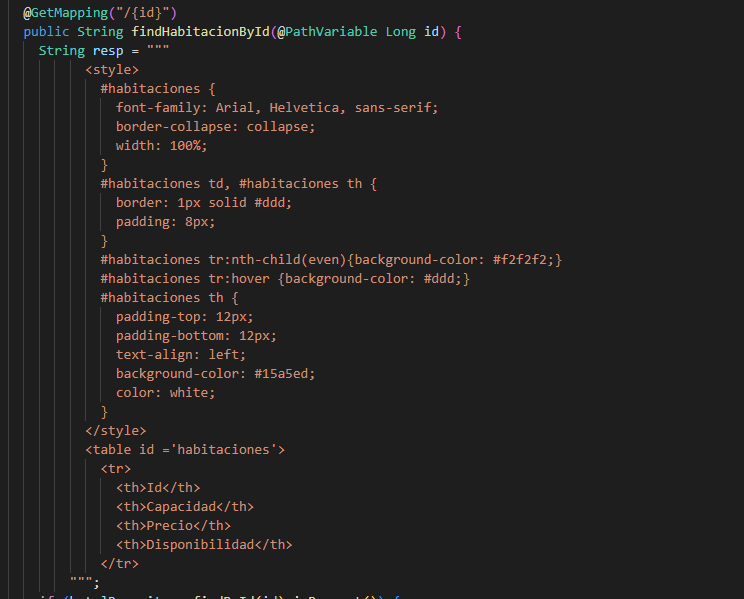
El primer error que cometimos fue escribir el nombre de la clase diferente de las tablas creada en phpMyadmin.

Hicimos modificaciones en el HTML cambiando color.

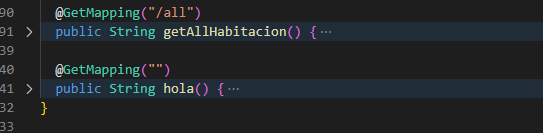
**Java\PROYECTO\Habitaciones\controllers\HabitacionController.java**

Esta es la clase controladora del proyecto, en ella se encuentran todos los métodos para crear los get, set y post del CRUD, para mostrar en el browser y hacer en el postman para crear, borrar y agregar un nuevo ID.



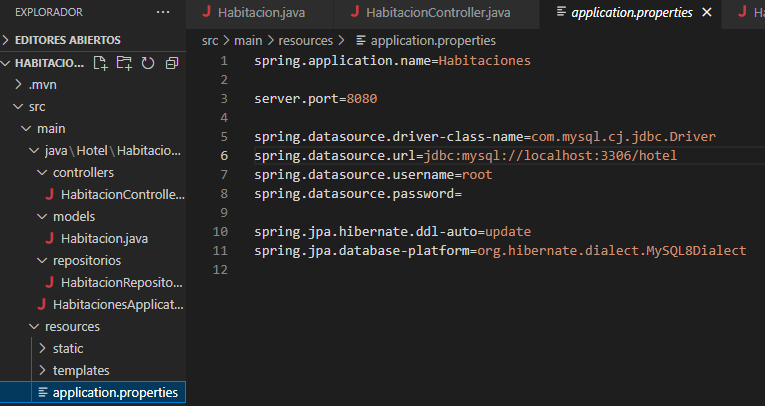






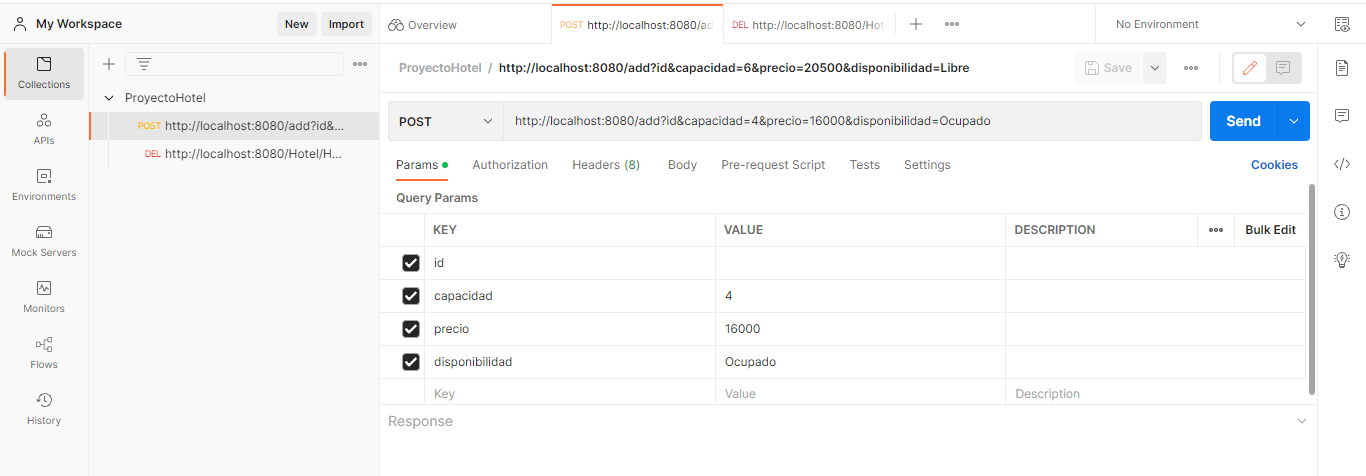
**EN LA CARPETA RESOURCES SE ENCUENTRA EL ARCHIVO APPLICATION.PROPERTIES:**

Esta incluye todo el código que nos permite unir la base de datos creada en mySQL con la aplicación.



**POSTMAN PARA AGREGAR Y BORRAR DATOS:**

Accedemos a Postman, una vez allí se eleige la opción a realizar (Post, Get, Delete,etc) con la respectiva url en este caso <http://localhost:8080/> + lo indicado en los get y post del código, en este caso se agrega “add”.



Una vez realizado este paso, se seleccionar SEND y el cambio se verá en el browser, como una nueva habitación en la lista generada:

