El modelo Reserva representa una reserva de una habitación en el hotel. Cada reserva tiene los siguientes atributos:

id: Un identificador único generado automáticamente para cada reserva.

nombreCliente: El nombre del cliente que realiza la reserva.

fechalnicio: La fecha de inicio de la estancia.

fechaFin: La fecha de finalización de la estancia.

tipoHabitacion: El tipo de habitación reservada

El sistema utiliza MySQL como base de datos. La tabla reservas almacena todas las reservas realizadas por los clientes. Los campos de la tabla coinciden con los atributos de la entidad Reserva y son gestionados automáticamente por Spring Data JPA.

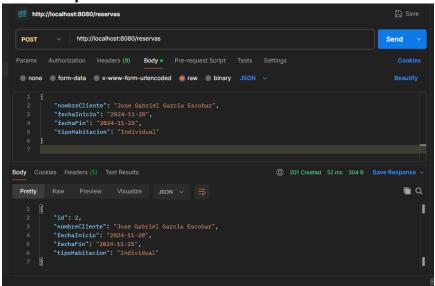
url mysql: localhost:3306/hotel\_reservas contraseña = contrasena123

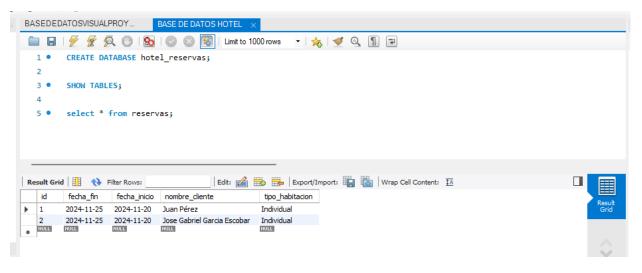
utilizarlo en el postman metodo get: localhost:8080/reservas

añadir en Headers: Content-Type, application/json

## Pruebas de que funciono:

## Postman pruebas:





```
DEBUG CONSOLE TERMINAL TEST RESULTS
                                                                                                                     袋 Run: HoteldataApplication 🕂 🗸 📗 🧃
                                  ))))
:: Spring Boot ::
                                 (v3.3.5)
2024-11-16T11:25:21.900-06:00 INFO 17832 --- [hoteldata] [
                                                                     main] data.hoteldata.HoteldataApplication
                                                                                                                   : Starting HoteldataApplication using Java 17
.0.13 with PID 17832 (C:\Users\Willys GaBo\Desktop\hoteldata\hoteldata\target\classes started by Willys GaBo in C:\Users\Willys GaBo\Desktop\hoteldata)
2024-11-16T11:25:21.904-06:00 INFO 17832 --- [hoteldata] [
                                                                     main] data.hoteldata.HoteldataApplication
                                                                                                                   : No active profile set, falling back to 1 de
fault profile: "default"
2024-11-16T11:25:22.872-06:00 INFO 17832 --- [hoteldata] [
                                                                     main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories
in DEFAULT mode.
2024-11-16T11:25:22.930-06:00 INFO 17832 --- [hoteldata] [
                                                                     main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in
51 ms. Found 1 JPA repository interface.
2024-11-16T11:25:23.649-06:00 INFO 17832 --- [hoteldata] [
                                                                     main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8080 (http)
2024-11-16T11:25:23.664-06:00 INFO 17832 --- [hoteldata] [
                                                                     main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2024-11-16T11:25:23.664-06:00 INFO 17832 --- [hoteldata] [
                                                                     main] o.apache.catalina.core.StandardEngine
                                                                                                                  : Starting Servlet engine: [Apache Tomcat/10.
```

```
vscode
 () launch.json
 () settings.json
hoteldata
                                                              import org.springframework.http.ResponseEntity;
                                                              import org.springframework.web.bind.annotation.DeleteMapping;

■ maven-wrapper.properties

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

✓ src

  ∨ main
                                                              import org.springframework.web.bind.annotation.PostMapping;

✓ java\data\hoteldata

    controller
                                                              import org.springframework.web.bind.annotation.RequestBody;
      import org.springframework.web.bind.annotation.RequestMapping;
                                                              import org.springframework.web.bind.annotation.RestController;

∨ model

      J Reserva.java
     repository

J ReservaRepository,java

     > service
                                                             @RestController
     J HoteldataApplication.java
                                                             @RequestMapping("/reservas")

∨ resources

                                                              public class ReservaController {
                                                                  private final ReservaService reservaService;
     > templates
     ■ application.properties
   > test
                                                                  public ReservaController(ReservaService reservaService) {
  target
                                                                      this.reservaService = reservaService;

    generated-sources\annotations

    generated-test-sources\test-annotations

                                                                  @PostMapping

∨ test-classes \ data \ hoteldata

                                                                  public ResponseEntity<Reserva> crearReserva(@RequestBody Reserva reserva) {
OUTLINE
```

```
日間で自

√ .vscode

                                                                                    import jakarta persistence GeneratedValue:
         {} launch.json

√ hoteldata

                                                                                    import jakarta.persistence.Table;
          .mvn\wrapper
           F maven-wrapper.properties
                                                                                    @Table(name = "reservas")
public class Reserva {
           ∨ main
∨ java\data\hoteldata
                                                                                         @Id
@GeneratedValue(strategy = GenerationType.IDENTITY)
                                                                                         private Long id;
              J ReservaController.java
                                                                                         @Column(name = "nombre_cliente", nullable = false)
private String nombreCliente;

J Reserva,java

٠
               J ReservaRepository.iava
                                                                                         @Column(name = "fecha_inicio", nullable = false)
private LocalDate fechaInicio;
0
              > service
                                                                                         @Column(name = "fecha_fin", nullable = false)
private LocalDate fechaFin;
              > static
              > templates
              F application.properties
                                                                                          private String tipoHabitacion:
                                                                                         // Constructor vacío para JPA
public Reserva() {}

√ target

y generated-sources \ annotations

                                                                                         public Reserva(String nombreCliente, LocalDate fechaInicio, LocalDate fechaFin, String tipoHabitacion) {
   this.nombreCliente = nombreCliente;
   this.fechaInicio = fechaInicio;

    generated-test-sources\test-annotations

             test-classes \ data \ hoteldata
       > OUTLINE
                                                                                               this.fechaFin = fechaFin:
                                                                                               this.tipoHabitacion = tipoHabitacion;
       > RUN CONFIGURATION
       > JAVA PROJECTS
```