

1. Que es la inyección de dependencia y representa los tipos de inyección de dependencia que existen.

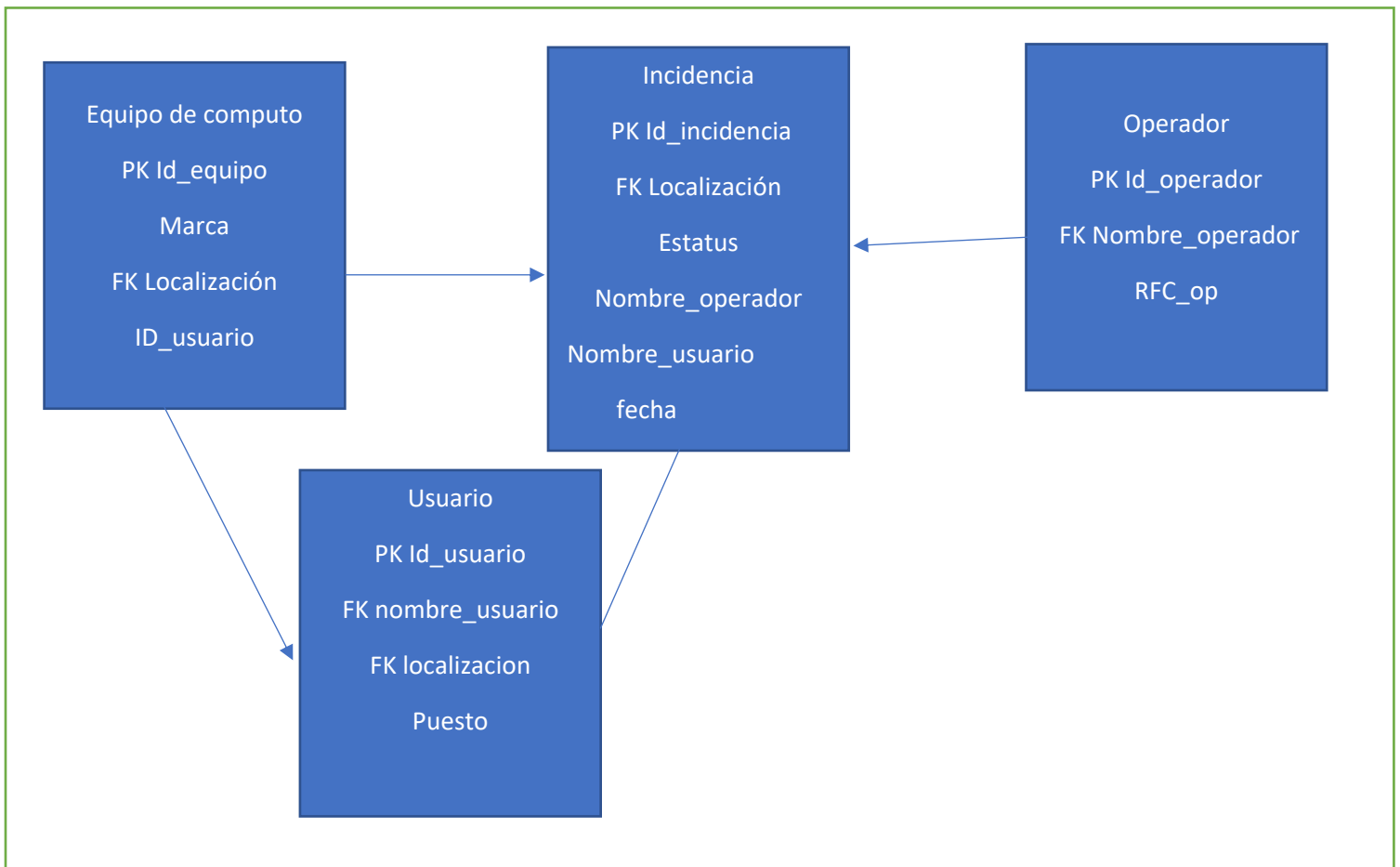
Es una serie de pasos o patrón de diseño que nos ayuda a crear instancias, esto con la finalidad de “prestar” clases que diferentes objetos dentro de otras clases necesiten, en el entorno de springboot

2. Menciona al menos 5 patrones de diseño y para que los utilizas.
 - a. Singleton: es uno de los métodos más usados y en la parte de creación, para asegurar que exista un punto de acceso global a una instancia, dentro de una clase
 - b. Proxy: este método es estructural y nos permite controlar las entradas a objetos
 - c. Decorador: a diferencia del proxy, este método que también es de la índole estructural dota de funcionalidades a las clases
 - d. Plantilla: genera una especie de molde, machote o esqueleto usando algoritmos para poder definir métodos o funciones, sin la necesidad cambiar la estructura del programa
 - e. Builder: construye objetos con una estructura definida con objetos más simples
3. Describe para que se utilizan las siguientes anotaciones;
@Autowired: funciona para inyectar las dependencias
@Qualifier: esta anotación nos va a ayudar a identificar cual es el bean o componente que se debe usar,
@PostConstruct: la utilizamos cuando queremos administrar la ejecución, uso y “destrucción” de un Bean después de hacerse la inyección de dependencia
@Required: la anotación indica que el componente debe estar completo o “lleno” con las propiedades que sean requeridas
@Bean: nos indica que es un componente que puede ser reutilizables, y lo identifica con un id
@Cacheable: habilitamos la memoria caché para que puedan ser almacenados datos o resultados

4. Que es un aspecto y describe un ejemplo de cómo lo podríamos implementar

Un aspecto define las funcionalidades transversales Se utiliza cuando existe una jerquia de clases, con la finalidad de. Evitar escribir el mismo código varias veces, esto es usado en estructuras complejas y poder separar responsabilidades

5. La empresa XYZ es la encargada de administrar el reporte de incidencias de los equipos de cómputo, donde un operador puede dar de alta una incidencia (“definir los datos que consideres más importante”), consultar incidencias por ID, por día, y por operador que las dio de alta, cabe mencionar que debemos tener dashboards que representen número de peticiones a los servicios consultados, insertados, cuantos se resolvieron, cuantos están abiertos. a. Realizar el diagrama E-R b. Realizar el diagrama de clases c. y programar un servicios rest de consulta e inserción que consideren mas importantes.



- Enlazar con el local host
- Enlazar con la base de datos (Mysql) desde properties
- Verificar los datos de build.gradle
- Realizar peticiones en la base de datos (SELECT * FROM.... WHERE, COUNT)
- Dropear tablas en la database antes de refrescar y levantar el servicio
- Para mostrar en pantalla hay que hacer uso del DOM
- Especifica en spring las relaciones manytoone, etc.

6. Que es la arquitectura de microservicios.

Fragmenta una aplicación en pequeñas tareas, para atomizar y hacer el proceso más eficiente comunicándose por medio de una API, dotando de independencia a la aplicación