

# TEMARIO MICROSERVICIOS CON SPRING CLOUD

- Instalación Java 8 o superior
- Instalación Maven
- Instalación MySQL
- Instalación Postgres
- Instalación STS
- Instalación Postman
- Microservicios
  - Crear un proyecto SpringBoot
  - Estructura de un proyecto springboot
  - Poner nombre a un servicio
  - Poner puerto a un servicio
- Hibernate
  - Anotaciones
    - @Entity
    - @Table
    - Propiedad name
    - @Id
    - @GeneratedValue
    - propiedad strategy
    - @Column
    - @Temporal
    - @Transient
- DAO - Repositorio
  - Clase CrudRepository
- Servicio
  - Anotacion @Service
  - Anotacion @Autowired
- Controlador
  - Anotacion @RestController
- Probando con POSTMAN
- Crear otro microservicio
- Crear Servicio para consumir rest (productos)
  - Clase AppConfig
  - Anotacion @Service
  - Implementar Servicio HTTPRESTTemplate
    - Metodo getForObject()
  - Crear Controlador
  - Libreria Feign para comunicar Microservicios
    - Activar Libreria OpenFeign
    - Anotacion @EnableFeignClients

- Anotacion @FeignClient
  - Parametro name
- Parametro url
- Anotacion @Qualifier
- Balanceo de Carga con Ribbon
  - Activar libreria Ribbon
  - Anotacion @RibbonClient
- Probar Balanceo de Carga
  - Utilizar objeto Enviroment
- Balanceo de Carga Ribbon con RestTemplate
  - Anotacion @Value
  - Anotacion @LoadValanced
- EUREKA Registro de Microservicios
  - Eureka Server
  - Dependencia Eureka Server
  - nombre de aplicacion
  - puerto de servidor
  - dependencia jaxb
  - Anotacion @EnableEurekaServer
- Conectar servicios como clientes en Eureka
  - Dependencia Eureka client
  - Anotacion @EnableEurekaClient
- Puerto dinamico
  - Expresiones regulares en application.properties
- Tolerancia de Fallos
  - Dependencia Hystrix
  - Anotacion @EnableCircuitBreaker
  - Lanzar una exception adrede
  - Anotacion @HystrisCommand
  - MEtodo alternativo a errores
- Timeout
- Servidor Zuul Api Gateway
  - Crear servidor
  - Configurar
  - Anotacion @EnableZuulProxy
  - Rutas dinamicas
- Filtros HTTP Zuul
  - Filtro pre, post, route
  - Clase ZuulFilter
- Configuración de Timeout en Zuul
- Spring Cloud Server
  - Crear Servidor
  - Anotacion @EnableConfigServer

- Instalar Git
  - Crear Repositorio local
  - Comando add
  - Comando commit
  - Configurar Git
- Leer Configuración
- Actualizar mocriservicio automatico
  - Anotación @RefreshScope
- Crear Repositorio Remoto GitHub
- CRUD Consumir Api
  - Anotación @RequestBody
  - Anotación @ResponseStatus
  - HTTP.CREATED
  - Anotación @DeleteMapping
  - Anotación @PutMapping
- Consumir API Rest
  - Utilizando RestTemplate
  - Usando Feign
- Servicio Commons
  - Crear .jar de Productos
  - Anotación @EntityScan
- Spring Cloud Security
  - JSON WEB TOKEN JWT
  - Introducción a OAuth2
- Microservicio Usuarios
  - Entidad Usuarios
  - Entidad Roles
  - Relación muchos a muchos
  - Crud en automatico
    - Usando @RepositoryRestResource
- Configurar servidor de autorización
  - Proceso de Login con OAuth2
- Configurar Zuul como Servidor de Recurso
- Configurar Cors Spring Security
- Eventos de éxito y fallo en autenticación
- Tres Intentos en Login
- Base de Datos MySQL
  - Configurar DataSource
- Base de Datos Postgres