

TEMARIO CURSO DE SPRING MVC

- Instalación JDK
- Instalación MySQL
- Instalación Eclipse
- Instalación Tomcat
- Integración Tomcat con Eclipse
- Hola Mundo con Spring
- Convertir proyecto a Maven
 - Explicación GroupId
 - Explicación ArtifactId
 - Estructura de un proyecto Maven
 - Dependencias
 - spring-core
 - spring-beans
 - spring-context
 - spring-aop
 - spring-web
 - spring-webmvc
 - spring-expression
- Archivo applicationContext
 - spring_config.xml
 - Cargar un archivo de configuración
- Inyección de Dependencia
 - Por Constructor
 - Etiqueta constructor-arg
 - Propiedad name
 - Propiedad value
 - Por Método setter
 - Etiqueta Property
 - Propiedad name
 - Propiedad value
 - Por referencia
 - Propiedad ref
- Anotaciones
 - Anotación @Autowired
 - Anotación @Component
 - Anotación @Value
- Creación de la Base de Datos
 - Dependencias para Base de Datos
- Configuración de Spring para Base de Datos
 - Bean DataSource

- Patrón de Diseño DAO
 - Paquete DAO
 - Paquete Pojo
- Insertar registros
 - Objeto NamedParameterJdbcTemplate
 - Método update()
 - Objeto MapSqlParameterSource
 - Método addValue()
- Consultar Registros
 - Objeto RowMapper
 - Método mapRow()
 - Objeto BeanPropertySqlParameterSource
- Consultas Personalizadas
 - Método queryForObject()
- Actualizar Registro
- Eliminar Registro
- Batch Updates
 - Objeto SqlParameterSourceUtils
 - Método createBatch()
 - Método batchUpdate()
 - Insertar Lista de Registros
- Transacciones
 - Configuración de TransactionManager
- Creación de un Proyecto WEB
 - Dependencias de un proyecto Web
- DispatcherServlet
 - Como funciona un DispatcherServlet
 - Configuración de un Prefijo
 - Configuración de un Subfijo
- Controlador
 - Anotación @Controller
 - Anotación @RequestMapping
 - Anotación @SessionAttributes
- Navegación entre Controladores
- Objeto Model
 - Anotación @ModelAttribute
 - Método addAttribute()
 - Obtener variables con JSTL
- Objeto SessionStatus
 - Método setComplete()
- SpringForms
 - Etiqueta sf:form
 - Propiedad action

- Propiedad method
- Propiedad commandName
- Etiqueta sf:input
 - Propiedad path
- Parametros entre Páginas
 - Anotación @RequestParam
- JNDI
 - Configuración de Tomcat
- Servicios DAO
- Mostrar Registros
 - Todos los registros
 - Anotación @PathVariable
- Mostrar registro en Formulario
- Guardar y Actualizar Registro
 - Método saveOrUpdate()
- Recursos
 - Carga de jQuery
- Eliminar Registro
- Redireccionar
 - Método addFlashAttribute()
 - redirect:/
- Integrar Hibernate a un proyecto Web con Spring
 - Dependencias de Hibernate
 - Guardar Registros
 - Actualizar Registros
 - Eliminar Registros
 - Configurar Hibernate con Spring
 - Fabrica de Sesiones de Hibernate
 - TransactionManager
 - Anotaciones JPA
 - Método save() de Hibernate
 - Consultar Registros con HQL
 - Consultar Registros con Criteria
- Anotación @Transactional
- Anotación @Repository
 - Relacionar 2 Formularios Web con Spring
- Spring Security
 - Dependencias
 - Filtro DelegatingFilterProxy
 - Filtro SpringSecurityFilterChain
 - Agregar Autenticación
 - Agregar Roles
 - Trabajando con Rutas

- Niveles de Acceso
- Denegar Acceso
- Personalizar Login
- Error de Logeo
- Spring Security TagLibs
 - Etiqueta sec:authorize
 - Etiqueta sec:authentication
 - Etiqueta security:logout
- Inicio de Sesión con Base de Datos
 - Creación de Tablas
 - Tabla authorities
 - Tabla usuarios
 - Objeto CustomAuthenticationProvider
 - Objeto CustomUserDetailsService
 - Formulario para dar de alta usuarios
- Encriptación de Contraseña
 - Bean passwordEncoder
 - Objeto PasswordEncoder
 - Método matches()
- Recordar Cuentas
 - Etiqueta security:remember-me
- Manejo de Errores
 - Personalizar Errores
 - Anotación @ExceptionHandler
- Agrupación de Validaciones
- JSON Autocomplete
 - Dependencias a utilizar
 - Anotación @ResponseBody
 - Propiedad produces