



# IFCT034PO. Desarrollo de aplicaciones con Java

**Sku:** PD\_IFCT034PO

**Horas:** 120

**Formato:** HTML

## OBJETIVOS

- Diseñar e implementar arquitecturas JAVA.
- Conocer los distintos tipos de datos que se pueden utilizar en Java.
- Crear clases de Java e instanciar objetos.
- Desarrollar proyectos en Java mediante el uso de entornos de desarrollo integrado o IDE.
- Manejar las estructuras de control y el flujo de programa.
- Utilizar las interfaces y las clases para conseguir el encapsulamiento y la abstracción de los datos.
- Aprender los conceptos de programación web con Java.
- Crear servlets de Java y páginas web a través de tecnología JSP.
- Manejar los Enterprise Java Beans.
- Comprender y utilizar el patrón MVC para la creación de aplicaciones más escalables.
- Utilizar PHP y su interoperabilidad con Java a través del puente Java Bridge.
- Aprender el funcionamiento básico de Struts.
- Crear proyecto Struts.
- Conocer los ficheros de configuración de Struts.
- Manipular las acciones y las vistas de los proyectos para añadir funcionalidades al proyecto.
- Utilizar las Struts Tags para crear formularios y mostrar los resultados.
- Aprender el funcionamiento avanzados de Struts.
- Crear diferentes tipos de acciones y resultados.
- Implementar las interfaces ModelDriven y Prepare.
- Manipular proyectos más complejos con la ayuda de clases de Modelo.
- Comprender el modelo DAO para obtener los datos a través de un enlace a la BBDD.
- Aprender el funcionamiento del framework Spring.
- Comprender el concepto de inyección de dependencias.
- Crear y manejar proyectos web con Spring MVC.
- Implementar Servicios web API Rest mediante Spring MVC.
- Utilizar el acceso a datos a través de JDBC API.

- Aplicar la capa de seguridad a nuestra aplicación.
- Aprender el funcionamiento del framework Hibernate.
- Crear entidades mediante el uso de anotaciones.
- Implementar el modelo DAO usando Hibernate.
- Conocer el mapeo de las asociaciones.
- Hacer uso de la herencia dentro del ORM.

## CONTENIDOS

- **Unidad 1: Programación orientada a objetos.**
  - Introducción a la programación orientada a objetos.
    - El lenguaje Java.
    - La máquina virtual de Java.
  - Variables.
    - Tipos primitivos.
    - Definición de variables.
  - Operadores en Java.
    - Operadores aritméticos.
    - Operadores relacionales y lógicos.
    - Operadores de asignación e incremento.
  - Precedencia de los operadores.
  - Sentencias de control.
    - Sentencias condicionales: la sentencia if.
    - Sentencias condicionales: la sentencia switch.
    - Sentencias iterativas: El bucle for.
    - Sentencias iterativas: los bucles while.
    - Sentencias iterativas: los bucles do while.
  - Clases.
  - Creación de una clase.
  - Objetos.
  - Los métodos.
    - Más sobre los métodos.
    - Control de acceso.
    - Los modificadores static y final.
  - Ejemplo práctico con objetos I.
  - Ejemplo práctico con objetos II.
  - Arrays.
  - Interacción entre los elementos.
  - Ejemplo de interacción entre elementos.
  - Interfaces I.
  - Interfaces II.
  - Peculiaridades de Java.
- **Unidad 2: Aplicaciones web con Java.**
  - Introducción, objetivos y software recomendado.
  - Programación web en Java. Introducción a JSP.

- Excepciones en Java.
  - Tipos de excepciones.
  - Excepciones definidas por el usuario.
  - Software de control de versiones – Git.
- Programación web en Java. Trabajo con Servlets.
  - Estructura de un servlet.
  - El ciclo de vida de un servlet.
  - Manejando datos de las peticiones I.
  - Manejando datos de las peticiones II.
  - Las cabeceras.
  - Los códigos de respuesta.
  - Compartiendo datos entre peticiones.
  - Páginas en JSP.
- Programación web con Java. Seguridad en JSP.
- Programación web en Java. Introducción a Enterprise Java Beans.
  - Uso de los JavaBeans con JSP.
- Programación web con JAVA. Introducción al patrón MVC.
- Programación web con PHP. Introducción a PHP.
  - Comenzando con PHP.
  - Variables en PHP.
  - Sentencias de control.
  - Funciones en PHP.
  - Orientación a objetos en PHP.
- Programación web con PHP. Mantenimiento de estado.
- Programación web con PHP. Servicios Web.
- Programación web con PHP. Herramientas adicionales.
- Configuración del servidor. Configuración del Servidor Resin.
- Configuración del Servidor. Instalación de Java Bridge.
- **Unidad 3: Java Struts 2.0 avanzado I.**
  - Introducción a Struts 2.
  - Patrón MVC y Beneficios.
  - Introducción a las Acciones.
    - Creación del primer proyecto.
  - Controlador con Servlet (Crear Vista).
  - Controlador con Servlet (Crear Servlet) I.
  - Controlador con Servlet (Crear Servlet) II.
  - Controlador con Servlet (Crear Acción).
    - Creando nuestro controlador en el proyecto.
  - Modificando las vistas.
  - Controlador con Filtro (Crear FilterDispatcher).
  - Fichero struts.xml.
  - Añadir Librerías Struts y crear Vista.
  - Crear Acción y fichero struts.xml.
  - Añadir Resultado a la acción.
  - Clase ActionSupport.
  - Creación de la Acción con ActionSupport.
  - Implementación de Interfaces de Contexto.

- Resultados de las Acciones.
- Cambiar resultados de las Acciones.
  - Flujo de Acción-Resultado.
- Struts tags.
- Object Graph Navigation language.
- Crear formulario con Form Tags.
- Crear Acciones.
  - Colecciones y tipos genéricos en Java I.
  - Colecciones y tipos genéricos en Java II.
- Acceder a la Pila de Valores.
- Creación de Acciones para Form Tags.
  - Usando los controles del formulario.
- Atributos List, ListKey y ListValue.
  - Crear un control select con ListKey y ListValue.
- Generics Tags.
- Crear Acción para Listado.
- Utilizar Generics Tags.
- Programar ordenación.
- **Unidad 4: Java Struts 2.0 avanzado II.**
  - Validación de entrada.
    - Crear el formulario.
    - Crear primeras validaciones.
    - Tipos de validación.
    - Añadir validadores al proyecto.
    - Etiquetas Set y Push.
    - Validación Personalizada.
    - Añadir lógica de validación.
  - Interceptores y conversión de tipos.
    - Crear el formulario.
    - Crear el conversor personalizado.
    - Interceptores ModelDriven y Prepare.
    - Implementación de ModelDriven.
    - Implementación de Prepare – Acción.
    - Implementación de Prepare - Struts.xml.
  - Autorización y autenticación.
    - Roles y usuarios.
    - Restringir acceso.
    - Definir autenticación.
    - Autenticación por formularios - Crear el formulario.
    - Mensajes e internacionalización. Añadir mensajes e internacionalización.
  - Capa de persistencia.
    - Preparar enlace a datos.
    - Implementación de la capa de persistencia.
    - Implementación del patrón DAO.
  - Subida/Descargas de ficheros.
    - Subida de ficheros y subida múltiple de ficheros.
    - Descarga de ficheros.

- Medidores de progreso.
    - Añadir medidor a un proceso.
  - AJAX.
    - Formulario con AJAX. Formulario con AJAX 2.
    - Freemarker, Velocity y Profiling.
    - Integración con Spring e Hibernate.
    - Migración desde versiones anteriores.
- **Unidad 5: Spring MVC.**
  - Introducción a Spring.
    - Características modular de Spring.
    - Inyección de dependencias I.
    - Inyección de dependencias II.
    - Preparación del entorno.
  - Creación de un proyecto Spring.
  - Spring Básico.
    - Inyección mediante anotaciones.
    - Formas de inyección.
    - Inyección mediante setter y atributos.
  - Spring MVC.
    - Primera aplicación con Spring MVC con Spring Boot.
    - Estructura del proyecto.
    - Creando el controlador.
    - Creando la vista.
    - Pasando datos a la vista.
    - Formularios en Spring.
    - Creando el formulario y la respuesta.
    - Validación de formularios: Modificando el controlador.
    - Validación de Formularios: Mostrando los errores en la vista.
  - Acceso a datos.
    - Implementando el patrón DAO I.
    - Implementación del patrón DAO II.
    - Creando los controladores para el acceso a datos.
    - Creando la vista para mostrar los datos.
    - Implementando la edición y eliminación.
  - Programación Orientada a Aspectos (AOP).
    - Desarrollando los conceptos de la AOP.
    - Ejemplo práctico de la orientación a aspectos.
  - Gestión de transacciones.
    - Implementando el uso de transacciones.
    - Uso de transacciones II.
    - Añadiendo el manejo de excepciones en la transacción.
  - Spring Web Services.
    - Implementando una API REST.
    - Implementando el controlador de la API Rest.
    - Más métodos del RestController.
  - Spring Security.
    - Implementando la seguridad.

- Implementando la seguridad: Defiendo los recursos.
- Spring Test.
  - Creando un test simple.
- Capacidades específicas.
- **Unidad 6: Hibernate.**
  - Introducción a Hibernate.
  - Hibernate.
    - Usando Hibernate en Spring.
    - Configurando la base de datos H2.
  - Clases mapeadas.
    - Resultado de la persistencia.
    - Creando el repositorio de datos.
    - Probando el repositorio de datos.
  - Consultas contra la base de datos.
    - Comprobando el resultado de la consulta.
    - @NamedQuery y @NamedQueries.
    - @NativeQuery y @NativeQueries.
  - Mapeo de asociaciones.
    - La relación uno a uno o One To One.
    - Haciendo uso de la relación.
    - Relación uno a muchos o muchos a uno.
    - Relación muchos a muchos.
    - Haciendo uso de la relación Mucho a Muchos.
  - Mapeo de Herencia.
    - Mapeo de Herencia. Creación de una nueva clase.
    - Formas diferentes de mapear la herencia.
  - Trabajo con objetos.
    - Manejando objetos.
    - Cambiando los estados de los objetos.
  - Optimización.
    - Sistemas de medida.
    - Estrategias de recuperación de asociaciones: Lazy y Eager