

ACCION FORMATIVA: JAVA, J2EE Y ARQUITECTURA DE NEGOCIO



CONTENIDOS

MODULO I. PROGRAMACIÓN EN JAVA

La tecnología Java

- Características claves de la tecnología Java.
- El Java Virtual Machine.
- El Garbage Collection.
- La seguridad en Java.
- Escribiendo compilando y corriendo aplicaciones Java.

Programación orientada a objetos con Java (un repaso)

- Clases.
- Objetos.
- Atributos.
- Métodos.
- Constructores.
- Inicializando objetos
- Usando los modificadores de acceso private y public.
- Encapsulando Objetos.
- Objetos relacionados.
- Herencia de clases.
- Agrupando clases con paquetes.
- Importando paquetes.

Fundamentos del lenguaje Java

- Archivos fuentes.
- Palabras reservadas e identificadores.
- Tipos de datos primitivos.
- Literales.
- Arrays.
- El método main y la inicialización de variables.
- Pasando argumentos.
- El Garbage Collection.

Operadores y asignación de valores

- Orden de evaluación de operadores.
- Operadores unarios.

- Operadores aritméticos.
- Los operadores Shift (<<, >> y >>>).
- Operadores de comparación.
- Operadores a nivel de bits (&, ^, y |).
- Operadores lógicos.
- El operador condicional (?)
- Operadores de asignación.

Modificadores

- Modificadores de acceso.
- Modificadores public y private.
- Modificadores Default y protected.
- Subclases y privacidad de métodos.
- Modificadores final y abstract.
- Modificadores e inicializadores static
- Modificadores native y transient
- Modificadores synchronized y volatile.

Conversión y Casting de tipos de datos.

- Cambios en tipos de datos implícitos y explícitos.
- Conversión de datos primitivos.
- Ascendencia de datos.
- Casting de datos primitivos.
- Conversión de objetos.
- Casting de objetos.

Control de flujo y excepciones.

- Las instrucciones while y do.
- La instrucción for.
- Las instrucciones break y continue.
- La instrucción if.
- La instrucción switch.
- Introducción a las Excepciones.
- Control de flujo y capturando Excepciones.
- Lanzando Excepciones.

Objetos y clases.

- Beneficios de la programación orientada a objetos.
- Tipos de datos abstractos.
- Reutilización.
- Implementando relaciones orientadas a objetos.

- Sobrecargando y sobrescribiendo métodos.
- Sobrecargando constructores.
- Introducción a las clases internas.
- La referencia this.
- Miembros de clases en relación a las clases internas.
- Clases definidas dentro de métodos.

Threads (programación multihilo).

- Introducción a los Threads.
- Ejecución y finalización de un thread.
- Estados y prioridades de los threads.
- Controlando Threads con la instrucción yield.
- Los métodos suspend y resume.
- El método sleep.
- El bloqueo de un Thread.
- Monitorear el estado de un Thread.
- Implementación del JVM para el control de threads.
- Los métodos wait y notify.
- Bloque de objetos y sincronización.
- El bloqueo de la muerte (Deadlock).

Los paquetes java.util y java.lang .

- La clase Object.
- La clase Math.
- Las clases Wrapper.
- La clase String.
- La clase StringBuffer.
- Las clases para colecciones.
- Tipos de colección.
- El uso de la clase Iterator.

Construyendo interfaces gráficas de usuario con Layout Managers.

- El paquete AWT y sus componentes.
- Contenedores y componentes.
- Componentes tamaños y posición.
- El Layout Manager.
- La ventaja del Layout Manager.
- El Flow Layout Manager.
- El Grid Layout Manager.
- El Border Layout Manager.
- El GridBag Layout Manager.
- Controlando el tamaño de la celda para un componente.
- Otras opciones de Layout.

Eventos.

- Manejando eventos de la interfaz gráfica de usuario.
- El modelo de delegación de eventos.
- Eventos semánticos y eventos de bajo nivel
- Observadores de eventos.
- Habilitación explícita de eventos.
- Adaptadores.
- Determinando la acción del evento.

Componentes.

- Métodos comunes de componentes.
- Los componentes visuales.
- Los componentes contenedores.
- Los componentes menu.

Archivos de texto; flujos de entrada y salida; y sockets.

- Argumentos de línea de comando y propiedades del sistema.
- Representaciones de texto y UTF.
- La clase File.
- Examinando y manipulando archivos y directorios.
- La clase RandomAccessFile.
- Leer y escribir texto dentro de archivos.
- Introducción a los Streams.
- Streams de bajo nivel.
- Streams de alto nivel.
- Readers.
- Writers.
- Encodings.
- Streams y serialización de objetos.
- Como crear una programa Java que trabaje sobre TCP/IP.
- Sockets.

DESARROLLO DE COMPONENTES WEB JAVA J2EE**Fundamentos de aplicaciones Web y el protocolo HTTP.**

- ¿Qué es una aplicación Web?
- Recursos pasivos y activos.
- Aplicaciones Web y el Web Application Server.
- Entendiendo el protocolo HTTP.

- La estructura de una petición HTTP.
- La estructura de una respuesta HTTP.

Fundamentos de Java Servlets.

- ¿Qué es un Servlet?
- ¿Qué es un contenedor de Servlets?
- Usando Tomcat.
- Escribiendo, compilando, instalando y corriendo un Servlet.
- Relación entre un Servlet container y el Servlet API.
- Los paquetes javax.servlet y java.servlet.http.
- Comparando el modelo Servlet con CGI y otras tecnologías Web del lado del servidor.

Fundamentos de JavaServer Pages.

- ¿Qué es una página JSP?
- Escribir e instalar una página JSP.
- Comparando el Servlet con una página JSP.
- Los modelos de arquitectura de una aplicación con JSP.

El modelo del Servlet

- Peticiones HTTP y el Web browser.
- Manejando las peticiones HTTP con la clase HttpServlet.
- Analizando la petición con las clases ServletRequest y HttpServletRequest.
- Enviando una respuesta con las clases ServletResponse y HttpServletResponse.
- El ciclo de vida del Servlet.
- La clase ServletConfig.
- La clase ServletContext.
- Compartiendo datos entre Servlets.
- Coordinando los servlets con la clase RequestDispatcher.

Estructura e implementación

- Directorio y estructura de una aplicación Web.
- El directorio WEB-INF.
- El archivo Web (WAR).
- El deployment descriptor.
- El elemento <servlet>.
- El elemento <servlet-mapping>.
- Asociando un URL a un Servlet.

El modelo del contenedor de Servlets.

- Inicializando la clase ServletContext.
- Manejando eventos.
- La clase ServletContextListener.
- La clase ServletContextAttributeListener.
- La clase HttpSessionAttributeListener.
- Configurando una aplicación Web.
- Aplicaciones Web en un entorno distribuido.

Manejo de excepciones en una aplicación Web.

- Manejando excepciones programáticamente.
- Manejando excepciones de lógica de negocios.
- Manejando excepciones declarativamente.
- Usando Servlets y páginas JSP como manejadores de excepciones.
- Usando la clase RequestDispatcher.

Manejo de sesiones.

- Fundamentos de estado y sesión.
- Trabajando con la clase HttpSession.
- Manejando eventos de sesiones.
- Invalidando una sesión.
- El tiempo de inactividad de una sesión.
- Soporte de sesiones usando cookies.
- Soporte de sesiones usando reescritura de URL.

Seguridad en una aplicación Web.

- Fundamentos de seguridad.
- Autenticación HTTP Basic.
- Autenticación HTTP Digest.
- Autenticación HTTPS Client.
- Autenticación FORM-based.
- Haciendo segura una aplicación Web de forma declarativa.
- El elemento display-name.
- El elemento web-resource-collection.
- El elemento auth-constraint.
- El elemento user-data-constraint.
- Haciendo segura una aplicación Web programáticamente.

Thread-Safe Sevlets.

- El modelo Multithreaded Servlet.
- El modelo Single-Threaded Servlet.
- Alcance de variables y Threads seguros.
- Alcance de variables locales, de instancia y de clase (estáticas).

- Alcance de atributos y Threads seguros.
- Alcance de Session, Context y Request.

El modelo JavaServer Pages.

- La sintaxis JSP.
- Directivas y declaraciones.
- Scriptlets, expresiones, acciones y comentarios.
- El ciclo de vida de una página JSP.
- Viendo la página JSP como un Servlet.
- La unidad de traducción.
- Atributos de directivas JSP.
- Los atributos import y session.
- Los atributos errorPage e isErrorPage.
- Los atributos language y extends.
- Los atributos buffer y autoflush.
- Los atributos isThreadSafe e info.
- Los atributos contentType y pageEncoding.

El modelo JavaServer Pages avanzado.

- El proceso de traducción.
- Variables JSP implícitas y objetos JSP implícitos.
- Entendiendo el alcance de una página JSP.
- Páginas JSP como documentos XML.

Componentes Web reusables.

- Inclusiones estáticas.
- Inclusiones dinámicas.
- Pasando parámetros y compartiendo objetos en inclusiones dinámicas.

Usando JavaBeans.

- Fundamentos de JavaBeans.
- JavaBeans serializados.
- Usando JavaBeans con acciones JSP.
- JavaBeans en Servlets.
- Accesando JavaBeans desde scriptlets JSP.
- Propiedades en JavaBeans.

Usando Custom Tags.

- Introducción a los Custom Tags
- Configurando al contenedor Web sobre la presencia de una librería de

- Custom Tags.
- Entendiendo la asociación explícita.
- Resolviendo los URIs con respecto a la ubicación de un TLD.
- Usando Custom Tags en páginas JSP.

Desarrollando librerías de Custom Tags.

- El Tag Library Descriptor.
- Los elementos del Tag Library Descriptor.
- El API para desarrollar Custom Tags.
- Implementando la interfaz Tag.
- Implementando la interfaz IterationTag.
- Implementando la interfaz BodyTag.
- Extendiendo la clase TagSupport y BodyTagSupport.
- Escribiendo Tags cooperativos.

Patrones de diseño para aplicación Web.

- Introducción a los patrones de diseño.
- Los patrones J2EE.
- El patrón Template.
- El patrón Model View Controller (MVC).
- El patrón Data Access Model (DAO).
- El patrón Business Delegate.
- El patrón Front Controller.

DESARROLLADOR DE COMPONENTES DE NEGOCIO ARQUITECTO J2EE

Introducción a la arquitectura de aplicaciones J2EE.

- Identificar tecnologías de múltiples capas.
- Evaluar la capa de negocios.
- El contenedor J2EE como middleware.
- Objetivo de los Enterprise JavaBeans.
- Beneficios de los Enterprise JavaBeans.
- Entendiendo la arquitectura del contenedor EJB.
- El Deployment Descriptor.

Arquitectura de Enterprise JavaBeans.

- Ejecutando un método remoto.
- Sobre los argumentos y los valores de retorno.
- La llamada a los métodos de negocio por medio de la interfaz remota.
- EJB usa RMI.
- El objeto remoto y su interfaz.
- Introducción a la arquitectura de los Session Beans.

- Introducción a la arquitectura de los Entity Beans.
- Introducción a la arquitectura de los Session Bean con estado.
- Introducción a la arquitectura de los Session Bean sin estado.
- Introducción a la arquitectura de los Message Driven Beans.
- Organizando sus Beans.

Session Bean.

- Entendiendo el cliente de un EJB.
- ¿Qué es JNDI?
- El método narrow.
- Escribiendo el Remote Home Interface para un Session Bean.
- Interfaces locales.
- Métodos de interfaces locales.
- El método remove.
- Comparando interfaces Remotas con interfaces Locales.

Detrás del Session Bean.

- Callbacks al contenedor.
- El ciclo de vida del Bean.
- La creación del Bean.
- Métodos de negocios de un Bean.
- Passivation: escalabilidad de un Session Bean.
- Removiendo un Bean.
- Escribiendo un Session Bean.
- El Session Context.

Los Entity Beans.

- ¿Qué es un Entity Bean?
- Los Entity Beans desde el punto de vista del cliente.
- Escribiendo un simple Entity Bean.
- Los Entity Beans y la interfaz remota.
- Los Entity Beans y el Home Interface.
- El cliente y el Home Interface.
- Métodos de negocios dentro del Home Interface.
- El método create de un Entity Bean comparándolo con el método create de un Session Bean.
- El método remove de un Entity Bean comparándolo con el método remove de un Session Bean.
- Ciclo de vida del Entity Bean.

Detrás de los Entity Beans.

- Sincronización de Entity Beans.
- Persistencia administrada por el contenedor y persistencia administrada por el Bean.
- La interfaz EntityBean y sus métodos Callback.
- Escribiendo un Entity Bean CMP.
- Identidad del Objeto: Primary Key.
- Los métodos Finder.
- Los métodos de negocio del Home Interface.

Relaciones entre Entity Beans.

- Relaciones.
- Multiplicidad.
- Campos CMP y CMR.
- Eliminación en cascada.
- EJB-QL para un simple bean.
- Las instrucciones SELECT y FROM.
- La instrucción WHERE.
- Colecciones.
- Las expresiones BETWEEN, IN, IS EMPTY, y LIKE.
- Asignación en las relaciones.

Message Driven Beans.

- Cuando usar Message Driven Beans
- Escribiendo un Message Driven Bean.
- El deployment descriptor para un Message Drive Bean.
- Topics y Queues.
- MessageDrivenContext.

Transacciones.

- ¿Cómo trabajan las transacciones con los EJB?
- Propagación de transacciones.
- Como lograr una transacción.
- El método setRollbackOnly.
- Transacciones administradas por el Bean (BMT).
- Transacciones administradas por el contenedor (CMT).
- Como trabajan los atributos.
- Inespecificado contexto de transacciones.
- El Deployment Descriptor para CMT.
- Sincronización de sesiones.

Excepciones en EJB.

- Manejando una excepción dentro de un EJB.
- Excepciones de aplicación dentro de un EJB.
- Excepciones de sistema dentro de un EJB.
- RemoteExcepcion y EJBExcepcion.
- Responsabilidades del Bean Provideer.
- Las cinco excepciones de aplicación estándares de los EJB.
- Comunes excepciones de sistema.

Seguridad en EJB.

- ¿Cómo trabaja la seguridad dentro de los EJB?
- La responsabilidad del Application Assembler.
- Definiendo la permisología de los métodos.
- La responsabilidad del Deployer.
- Seguridad a nivel de instancias vs. seguridad a nivel de clases.
- Usando seguridad programática.
- Uso de <run-as>
- El contexto de la seguridad y su propagación con <run-as>

Distribuyendo EJB.

- El contexto java:comp/env.
- Creando y usando subcontextos.
- La responsabilidad del Bean Provideer y del Application Assembler.
- La responsabilidad del Deployer.
- Recordando las responsabilidades.
- El API de los EJB.
- El EJB-JAR.
- Programando restricciones.