



¿Qué es Java?

La tecnología Java aporta las herramientas necesarias para desarrollar aplicaciones mediante la Programación Orientada a Objetos con independencia del sistema operativo en el que se trabaje.

¿En qué consiste el Master en Java de Artes?

En el Master en Java el alumno podrá aprender desde la metodología de la programación (incluyendo un punto importantísimo como es la programación orientada a objetos), pasando por el estudio avanzado en varias de las tecnologías que ofrece Java (RMI, JFC, AWT) hasta el desarrollo de aplicaciones web bajo la plataforma J2EE (Java™ 2 Platform Edition) con servlets, JSP y EJB.

Ventajas del Master de Java sobre cualquier tipo de manuales o tutoriales de Java

Los Masters en Java online de Artes le permitirán profundizar en los conocimientos de esta materia gracias a las nuevas técnicas de e-learning que facilitan el aprendizaje, a diferencia de un simple tutorial de Java en el que se hace más complicada la asimilación. Esta diferencia con otro tipo de cursos o con cualquier manual de Java es particularmente notable en el caso de cursos personalizados, en los que el alumno recibe asistencia del tutor de Artes durante todo el curso.

Duración del curso

El alumno puede comenzar el curso 24 horas después de formalizada la matrícula. La duración del mismo será de 6 a 9 meses (dependiendo de la capacidad del alumno) aprox. (dependiendo de la capacidad del alumno) o a convenir con el alumno si se elige la modalidad "curso personalizado".

A continuación detallamos los módulos del Master en Java (J2SE-J2EE):

1º) Módulo: Metodología de la Programación

1. Introducción a la Programación
 1. Historia de la informática <LICONCEPTOS< li>
2. Introducción a la lógica matemática
 1. Lógica booleana
 2. Operadores lógicos
 3. Tablas de verdad
 4. Tautologías, contradicciones
3. Representación abstracta de código
 1. Pseudocódigo
 2. Diagramas

4. Sintaxis de Pascal
 1. Sintaxis
 2. Glosario de palabras reservadas
 3. Freepascal
5. Definiciones de datos
 1. Tipos de datos
 2. Variables y constantes
 3. Subrangos y enumerados
6. Operadores
 1. Operadores aritméticos
 2. Operadores de asignación
 3. Operadores de comparación
 4. Operadores lógicos
 5. Precedencia de operadores
7. Estructuras complejas de datos
 1. Tipos de datos definidos por el usuario
 2. Registros
 3. Conjuntos
 4. Vectores y matrices
8. Estructuras de Control Selectivas
 1. If...then...else
 2. Case
9. Estructuras de Control Repetitivas
 1. For
 2. While
 3. Repeat
10. Modularización
 4. Procedimientos
 5. Funciones
 6. Estructura de una UNIT
 7. Sentencia USES
10. Listas
 1. Implementación con punteros
 2. Listas enlazadas
 3. Recorrido de listas con bucles
11. Recursividad
 1. Factorial
 2. Recorrido de listas

12. Bases de Programación Orientada a Objetos

1. Introducción
2. Objetos
3. Clases
4. Métodos
5. Herencia

2º) Módulo: Introducción a la Plataforma Java

1. Introducción

1. Origen del Java
2. Características principales

2. Entorno Java

1. JDK: Arquitectura e Instalación (variables de entorno, etc.)
 - Práctica: Instalación del J2SDK 1.4.
2. El compilador javac
3. El entorno de ejecución java
4. El entorno de ejecución de applets appletviewer
5. Entornos de desarrollo Java (Jbuilder, Forte4Java, etc.)

3. El lenguaje Java

1. Sintaxis
2. Herencia, clases e interfaces, arrays, etc.
3. Operadores, control de flujo, precedencia, etc.
4. Aplicaciones básicas en Java.
 - Ejercicio: Primera aplicación de ejemplo en Java

3º) Módulo: POO

1. Introducción a POO

1. Introducción a la orientación a objetos.
2. Objetos en Java.
3. Clases en Java.
4. Clases Java de utilidad

2. Características de la orientación a objetos

1. Abstracción.
2. Herencia: subclasses y superclases, la Clase Object.
3. Polimorfismo: Sobrecarga de métodos.
4. Ventajas de la orientación a objetos

3. Estructuras de datos complejas
 1. Introducción.
 2. Listas (pilas y colas)
 3. Árboles
 4. Grafos
4. La eficiencia de los algoritmos
 1. Notación asintótica.
 2. Reglas para el cálculo de la eficiencia
 3. Eficiencia en algoritmos recursivos
5. Divide y vencerás
 1. Características generales
 2. Ordenación
 3. Ordenación por mezcla (Mergesort)
 4. Ordenación rápida (Quicksort)
 5. Editor ortográfico avanzado: función ordenar
6. Algoritmos Voraces
 1. Características generales
 2. Problema de la mochila
 3. Árboles de recubrimiento mínimo
 - Algoritmo de Kruskal
 - Algoritmo de Prim
7. Exploración de árboles
 1. Recorrido de árboles
 2. Recorrido en profundidad
 3. Recorrido en anchura
 4. Backtracking
 5. Ramificación y poda
 6. Minimax
8. Backtracking
 1. Esquema general
 2. Ejemplo: búsqueda en un árbol binario
 3. Editor ortográfico avanzado: función verificar ortografía
9. Introducción a los patrones de diseño
 1. Concepto de patrón de diseño
 2. Patrones relacionados con la creación de objetos
 3. Patrones estructurales
10. ¿Qué es UML?
 1. Un paseo por UML

4º) Módulo: J2SE

1. Applets

1. Características de los Applets
2. Importar clases
3. El ciclo de vida del Applet
4. Crear el primer Applet
5. Insertar un Applet en una página HTML
6. Parámetros en un Applet

2. E/S en Java (ficheros)

1. Introducción a los Streams
2. Cómo usar Streams en ficheros
3. Usar Piped Streams
4. Concatenar ficheros.
5. DataInputStream y DataOutputStream (ficheros serie)
6. Trabajar con ficheros de acceso aleatorio
7. Escribir filtros para ficheros de acceso aleatorio

3. Excepciones

1. Manejo de errores utilizando excepciones.
2. ¿Qué es una excepción y por qué debo tener cuidado?
3. Requerimiento de Java para las excepciones
4. Capturar y manejar excepciones.
5. Especificar las excepciones lanzadas
6. La sentencia throw
7. La clase Throwable y sus subclases
8. Crear clases de Excepciones.

4. Threads

1. ¿Qué es un thread?
2. Atributos de un thread
3. El cuerpo de un thread
 - Ejercicio: El applet del reloj digital
4. El estado de un thread
5. Prioridad de un thread
6. Threads servidores (daemons)
7. Grupos de Threads
8. Programas Multi-threads
9. Sincronización de threads
10. Monitores Java
11. Los métodos wait() y notify()

5. AWT

1. Descripción de los interfaces de Usuario
2. Estructura del AWT
3. Componentes
 - Distintos componentes button, label, menu, etc.
4. Contenedores
 - Distintos paneles, window, frame, etc.
5. Layouts (distribuciones)
 - Distintos layouts, BorderLayout, GridLayout, etc.
6. Manejo de Eventos en AWT.
7. Gráficos en el AWT (clase Graphics)
 - Distintos elementos gráficos rectangle, etc.
 - Dibujar texto en el AWT clase FontMetrics, etc.

6. SWING - JFC

1. Descripción de la Java Foundation Classes
2. Estructura Swing
3. Componentes
 - Distintos componentes JButton, JLabel, JMenu, etc.
4. Contenedores
5. Layouts (distribuciones)
 - Distintos layouts, BorderLayout, GridLayout, etc.
6. Manejo de Eventos en Swing.
 - Oyentes de eventos en swing
 - Diferentes tipos de oyentes, ActionListener, DocumentListener, etc.

7. RMI - Métodos Nativos

1. Trabajar con JNI
2. Paso a paso
3. Ejecutar Java
4. JNI

8. Trabajo en Red con Java

5. Trabajo en red básico
6. URLs
7. Sockets.
8. Datagramas
9. Controladores de seguridad.

5º) Módulo: J2EE

1. Prefacio (Software Necesario)
 1. Obtener e Instalar el Software Necesario.
 - Ejercicio: Instalación y Prueba del software.
 2. Herramientas que Componen la Plataforma J2EE.
2. Componentes Web
 1. Ciclo de Vida de los Componentes Web
 2. Archivos de Aplicación Web
 - Práctica: Crear un fichero WAR
 3. Descriptor de Despliegue
 - Ejercicio: Distinguir elementos de un descriptor de despliegue
 4. Desplegar una Aplicación Tomcat
 - Práctica: Desplegar tu primera aplicación.
3. Servlets
 1. Introducción a los Servlets
 2. Estructura básica de un Servlet
 - Ejercicio: El servlet 'HolaMundo'
 3. Manejar Formularios en un Servlet
 - Ejercicio: Crear un Descriptor de despliegue.
 4. Cabeceras de Solicitud en Servlets
 5. Variables CGI en Servlets
 - Práctica: Leer variables CGI.
 6. Códigos de Estado HTTP
 7. Especificar Cabeceras de Respuesta HTTP
 8. Manejar Cookies
 - Ejercicio: Buscar Clases necesarias
 9. Seguimiento de Sesión
 - Ejercicio: Obtener información de la sesión.
4. JavaServer Pages (JSP)
 1. Introducción a las Páginas JSP
 2. Scriptles JSP
 - Ejercicio: Saludo
 3. Directivas JSP
 4. Acciones JSP
 5. Manejar Formularios con JSP
 - Práctica: El ejemplo 'Hello' con JSPs
 6. Manejar Excepciones en JSP
 - Ejercicio: lanzar una página de error.
 7. Librerías de etiquetas JSP personalizadas
 - Ejercicio: Aplicación utilizando etiquetas personalizadas
 8. Apéndices - JavaServer Pages

5. JavaBeans Enterprise

1. Introducción
2. Qué es un JavaBean Enterprise
3. Qué es un Bean de sesión
4. Qué es un Bean de entidad
5. Definir los Accesos de clientes con interfaces
6. Contenidos de un Bean Enterprise
7. Convenciones de nombrado para Bean Enterprise
8. El ciclo de vida los Beans enterprise
9. Práctica: Crear un Bean de sesión
10. Ejercicio: Crear un Bean con persistencia controlada por el Bean
11. Ejercicio: Crear un Bean con persistencia controlada por el contenedor

6. JDBC (Acceso a Bases de Datos)

1. Arquitectura de una Aplicación
2. Drivers JDBC (varios tipos)
 - Ejercicio: Instalación de un driver JDBC.
3. Transacciones
 - Ejercicio: Transacciones
4. Fuentes de datos (DataSource, etc.)
5. Sentencias preparadas
 - Ejercicio sobre sentencias preparadas.
6. Tipos de datos avanzados
7. Bases de datos en Red
 - Ejercicio: Utilización de JDBC con servlets
 - Ejercicio: Utilización de JDBC con JSPs.
 - Práctica: Utilización de JDBC con Servlets, JSPs y JavaBeans Enterprise

7. Proyecto Fin de Curso

1. Diseñar las tablas de la base de datos.
2. Diseñar el modelo a utilizar MVC.
3. Diseñar el Inteface de usuario (JSPs)
4. Diseñar los JavaBeans y Servlets Necesarios
5. Seguridad, definir los roles