

# Roadmap

## Índice

1 Aspectos destacados.....	2
2 Certificación Sun.....	3
3 Recursos adicionales.....	4
3.1 Bibliografía.....	4
3.2 Enlaces.....	4

## 1. Aspectos destacados

- Manejo de las herramientas java, javac, jar
- Creación de proyectos Java en el entorno de desarrollo Eclipse
- Configuración y manejo del entorno Eclipse
- Organización del directorio de desarrollo
- Sintaxis básica del lenguaje Java
- Interfaces y herencia
- Normas de estilo para el código fuente Java
- Automatización de la construcción y despliegue de aplicaciones con Ant
- Tratamiento de errores mediante la captura de excepciones
- Lanzar excepciones para la gestión de errores
- Identificar los casos en los que es necesario la utilización de hilos
- Creación de hilos mediante herencia y mediante interfaces. Determinar la forma más adecuada en cada caso
- Operaciones básicas y propiedades de los hilos
- Métodos para la sincronización de hilos. Identificar los casos en los que es necesario utilizar cada una de ellas
- Manejo de las operaciones básicas de los flujos de E/S
- Tipos de hilos existentes y funciones de cada uno
- Acceso a ficheros mediante hilos de E/S
- Acceso a los recursos de una aplicación
- Manejo de los flujos DataInputStream y DataOutputStream
- Creación de proyectos Java en el entorno de desarrollo Eclipse
- Mecanismo de serialización de objetos
- Acceder a una URL desde una aplicación Java
- Programación del protocolo HTTP desde una aplicación cliente
- Visión general de los componentes principales para crear aplicaciones AWT/Swing (botones, menús, desplegados, ventanas, etc)
- Uso de gestores de disposición (*layout managers*) para colocar los componentes en aplicaciones gráficas AWT/Swing
- Manejo de eventos de diferentes tipos (de acción, de ratón, de ventana, etc) en aplicaciones AWT/Swing
- Concepto de Applet, programación y uso de los mismos. Similitudes y diferencias con aplicaciones AWT/Swing *standalone*
- Tener claro cómo se gestiona la seguridad en aplicaciones Java: qué elementos intervienen en dicha gestión, y qué restricciones se aplican (o se deben aplicar) a los Applets
- Uso de opciones avanzadas en el desarrollo de aplicaciones Swing: acciones, bordes, timers, etc
- Creación de casos de prueba en JUnit e identificación de los métodos de prueba a implementar
- Prueba de los métodos y casos de prueba desde Eclipse

- Funcionalidades adicionales en los casos de prueba: método `setUp`, pruebas con excepciones, creación de las clases tras los casos de prueba, etc
- Estructuración de casos de prueba en árboles mediante suites de prueba
- Uso del depurador de eclipse: establecimiento de breakpoints, ejecución paso a paso, inspección de valores de variables
- Componentes principales de Log4Java: loggers, appenders y layouts
- Inserción de mensajes de log en nuestras aplicaciones Java
- Configuración del logging mediante ficheros de properties y mediante ficheros XML
- Uso de commons-logging para encapsular Log4J

## 2. Certificación Sun

Aunque el resto del curso de Especialista está orientado a certificaciones en el ámbito de J2EE, este módulo de Java y Herramientas de Desarrollo se basa en las certificaciones para Java básico o estándar. Dichas certificaciones son dos:

- **SCJP** (*Sun Certified Java Programmer* - Certificado Sun de Programador Java): <http://www.sun.com/training/certification/java/scjp.xml>
- **SCJD** (*Sun Certified Java Developer* - Certificado Sun de Desarrollador Java): <http://www.sun.com/training/certification/java/scjd.xml>

Para la primera (**SCJP**), se debe completar un examen. En el caso del certificado para la plataforma Java 1.5, el examen abarca los siguientes apartados generales:

- **Sección 1: Declaraciones, inicializaciones y ámbitos:** evalúa si el alumno es capaz de escribir código que declare clases o interfaces, utilice adecuadamente la estructura de paquetes e imports, utilice código con tipos simples, arrays, objetos estáticos, variables locales, constructores, métodos estáticos y no estáticos, sobrecarga de métodos, etc
- **Sección 2: Control de flujo:** uso de sentencias `if`, `switch`, bucles (`for`, `do`, `while`, `break`, `continue`), manejo y uso de excepciones (`try-catch-finally`), etc
- **Sección 3: Contenidos del API:** uso de wrappers básicos (`Integer`, `Boolean`, etc), entrada/salida de ficheros, serialización de objetos para E/S, formateo de datos con el paquete `java.text`, y parseo de cadenas mediante expresiones regulares y similares (paquetes `java.util` y `java.util.regex`)
- **Sección 4: Concurrencia:** manejo de hilos (mediante `Thread` y mediante `Runnable`), estados de los hilos, interbloqueos y sincronización
- **Sección 5: Conceptos sobre orientación a objetos:** desarrollo de código que cumpla los requerimientos de encapsulamiento, cohesión y acoplamiento entre clases (mucho cohesión, poco acoplamiento). Uso del polimorfismo y del casting. Uso de métodos sobrecargados, llamadas a la superclase, etc.
- **Sección 6: Colecciones:** determinar qué tipos de colecciones (listas, hashmaps, etc) utilizar en diferentes supuestos. Comparaciones y ordenaciones entre objetos de una colección, etc
- **Sección 7: Fundamentos:** uso correcto de los modificadores de acceso, declaraciones

de paquetes, imports. Seguimiento de trazas. Manejo de referencias a objetos. Uso del recolector de basura... etc

Para la segunda (**SCJD**) es necesario haber obtenido antes la primera certificación (SCJP). Después, se deben superar dos pruebas: un supuesto de programación, y un examen.

La primera prueba (el supuesto de programación), consiste en escribir código para implementar una supuesta aplicación para empresa. Se evaluarán aspectos como documentación, diseño orientado a objetos, desarrollo de la interfaz gráfica, interbloqueos, etc.

La segunda prueba (el examen), es una explicación sobre el desarrollo que hayamos hecho en el supuesto de programación anterior, explicando las decisiones principales que hemos tenido que tomar, ventajas y desventajas de las mismas, y justificación de dichas decisiones, en función de los objetivos propuestos para la implementación.

Más información sobre las certificaciones en los enlaces respectivos vistos antes.

### 3. Recursos adicionales

---

#### 3.1. Bibliografía

---

- **Curso de Java**, Ian F. Darwin, *Ed. Anaya Multimedia, Colección O'Reilly*
- **Java 2 v5.0**, Varios autores, *Ed. Anaya Multimedia, Colección Wrox*
- **Piensa en Java**, Bruce Eckel, *Ed. Prentice Hall*
- **Core Java 2**, Cay Horstmann y Gary Cornell, *Ed. Prentice Hall PTR*
- **Java in a Nutshell**, David Flanagan, *Ed. O'Reilly Media*
- **Professional Java Tools for Extreme Programming: Ant, XDoclet, JUnit, Cactus, and Maven**, Richard Hightower, Warner Onstine, Paul Visan, Damon Payne y Joseph D. Gradecki, *Ed. Wrox*
- **JUnit in Action**, Ted Husted, Vincent Massol, *Ed. Manning Publications*

#### 3.2. Enlaces

---

- [Web oficial de Java, http://www.java.sun.com](http://www.java.sun.com)
- [Web oficial de Eclipse, http://www.eclipse.org](http://www.eclipse.org)
- [Web del proyecto JUnit, http://www.junit.org](http://www.junit.org)
- [Web de Apache Ant, http://ant.apache.org](http://ant.apache.org)

*Roadmap*