

1

Ejercicios prácticos

**Unidades**: 3

# Guía

* Se realizarán varios ejercicios relacionados con la preparación del entorno de Java. Por cada ejercicio existen unas preguntas a responder.
* Cada ejercicio esta pensado en poder realizarlo con apoyo del temario, si es insuficiente, recurrir a buscar información por internet.
* La práctica está orientada para realizar su desarrollo en un sistema operativo de Windows. Se puede realizar en cualquier otro sistema, pero puede cambiar su procedimiento.
* Los requerimientos necesarios son:
  + Disponer de espacio suficiente en el disco duro
  + Tener permisos de instalación
  + Mínimo 512 MB de memoria RAM

# Ejercicio 1. Instalación de la JDK de java.

Procedimiento

1. Descargar de la página oficial de Oracle, la última versión disponible de la JDK para Windows
2. Proceder con la instalación y verificar que se instala correctamente.
3. Buscar la ruta donde se ha instalado.
4. Revisa las variables de entorno PATH y CLASSPATH. Se puede acceder a las variables de entorno mediante:

Panel de control/Sistema y seguridad/Sistema/Configuración avanzada del sistema

En el Path debe de aparecer un registro que apunta a la ruta donde esta instalado la JDK

Responde a las siguientes preguntas

* + ¿Qué diferencia existe entre la JDK y la JRE?

El JDK es el entorno de desarrollo utilizado para construir aplicaciones basadas en java, y es suministrado por Oracle. El JRE, por su parte, es el entorno de ejecución, un conjunto de bibliotecas o librerías que permiten un tratamiento del bytecode o código intermedio entre el código fuente y el código máquina.

¿Como programadores que entorno necesitamos la JDK o JRE?

Necesitamos ambos, la JDK para el desarrollo de las aplicaciones java, y el JRE para su ejecución

¿Y si solo voy a ejecutar programas en java?

Sería necesario contar con el JRE.

* + Existe un entorno denominado OpenJDK. Busca porque existe una versión denominada OracleJDK y otra OpenJDK

Ambas son Java Development Kits. La principal diferencia es que OpenJDK es un proyecto open source mantenida por Oracle, Red Hat y la comunidad, mientras que OracleJDK es un proyecto de código cerrado y requiere licencia de pago, cuyo mantenimiento corre únicamente a cuenta de Oracle

* + ¿Por qué es necesario poner la ruta de la instalación en el PATH?

Porque permite al sistema indicar la localización exacta donde se encuentra el ejecutable de un software o programa de cara a su ejecución por comandos o desde la terminal.

* + Abre el símbolo del sistema (Inicio y escribir cmd o símbolo del sistema) y ejecutar el siguiente comando *java -version.* ¿Qué resultado da? Nota si aparece un error revisar de nuevo el punto 4. de la instalación

Java 20.0.2 2023-07-18

1

# Ejercicio 2. Ejecución de una aplicación sin IDE

Procedimiento

1. Crear una carpeta llamada *Ejercicios* dentro de C:
2. Crear un fichero de texto llamado *Ejercicio2.java* dentro de la carpeta del paso 1. Es importante que la primera letra este en mayúsculas.

*Nota: Si tenéis oculto que muestre las extensiones protegidas por el sistema operativo, os creará un fichero tal que así Ejercicio2.java.txt. Se debe habilitar que el sistema operativo muestre las extensiones y borrar del nombre la extensión*

*.txt*

1. Abrir el fichero con el bloc de notas y añadir el siguiente código:

**import** java.time.LocalDateTime;

**import** java.time.format.DateTimeFormatter;

**import** java.util.Scanner;

**public class** Ejercicio2 {

**public static void** main(String[] args) {

DateTimeFormatter dtf = DateTimeFormatter.*ofPattern*("dd/MM/yyyy"); LocalDateTime fecha = LocalDateTime.*now*();

System.***out***.println("Hoy es : " +dtf.format(fecha) + ", un fantastico día!"); System.***out***.println("Indicamente como te llamas:");

Scanner entrada = **new** Scanner(System.***in***); String contenido = entrada.next();

System.***out***.println("Un placer de conocerte " + contenido);

}

}

1. Abrir el símbolo del sistema (Inicio y escribir cmd o símbolo del sistema) y ejecutar: *cd C:\Ejercicios* y nos ubicará en la carpeta que habíamos creado. Ahora ejecutar un segundo comando: *java Ejercicio2.java*

Responde a las siguientes preguntas:

* + Texto

    Descripción generada automáticamenteMuestra una captura de pantalla que se pueda ver el resultado de la ejecución del último programa.
  + Explica que hace el comando *System.out.println*

El comando System.out.println devuelve como salida por consola una cadena de caracteres que se introduce previamente en el código. Puede establecerse directamente el String o puede mandarse a imprimir por consola una variable determinada.

* + Ejecuta el comando: *javac Ejercicio2.java* y mirar en la carpeta, ¿Ha creado un nuevo fichero? Explica para que sirve.

Dicho comando permite la conversión del código fuente a código intermedio o bytecode, generando un archivo .class, que será ejecutable en el JRE.

# Ejercicio 3. Instalación de un entorno de desarrollo

Existen diferentes entornos de desarrollo para Java, busca información sobre Eclipse y Netbeans, descárgate ambos y procede a instalarlos.

Responde a las siguientes preguntas:

* + ¿Cuál te gusta más? Indica el por qué
  + ¿Qué diferencia encuentras entre trabajar con un entorno gráfico o no hacerlo?
  + A parte de Eclipse y Netbeans, ¿Existen otros entornos gráficos? Mencionalos.
  + Crea un proyecto de cero, explica como lo has hecho.
  + Existen diferentes tipos de proyecto. Para una aplicación normal de java. ¿Qué tipo de proyecto tenemos que crear?

# Ejercicio 4. Ejecución de una aplicación con IDE

Procedimiento

1. Crea un proyecto de cero
2. Crea una clase y nómbrala *Ejercicio4*. Introduce el código que hay del ejercicio 2 Responde a las siguientes preguntas:
   * Al finalizar el paso 2, ¿El IDE os muestra alguna advertencia o error? Si es así, explica que esta sucediendo y como solucionarlo.
   * Investiga como se ejecuta el programa y muestra una captura de pantalla con el resultado.
   * ¿Por qué dentro de la clase existe un método llamado *main*?¿Pueden existir más de uno en diferentes clases?

# Ejercicio 5. Ha llegado tu turno.

Realiza un pequeño programa que pida dos números, realice su división y muestre el resultado. Realizar primero el pseudocódigo para que os sea más fácil. Realizar algunas pruebas de ejecución y probar con estos datos número 1 = 5, número 2 = 0. ¿Qué sucede?

¿Cómo lo puedes solucionar?

La consola devuelve un error exception ya que no admite la división entre 0 de enteros. Solución: Contemplar una condición por la cual si el usuario introduce como segundo número el valor de 0, se devuelva una cadena de caracteres con resultado 0, siendo este también parte del String.