Práctica de Java: Características, compilación, máquina virtual

Nome e apelidos:

SECCIÓN 1: VARIABLE DE ENTORNO "PATH"

No teu ordenador abre un terminal[1].

Se é Windows, executa o comando `echo %PATH%`
Se é Linux, execute o comando `echo $PATH`

1.1. Cal é a saída que ves?

1.2. Que significa? (Explícao coas túas propias palabras)

🧠 Axúdate da túa IA favorita (Claude, Copilot, ChatGPT,…) para obter respostas se o precisas. Interioriza a información e responde coas túas palabras.

En base a iso, **identifica** algún executable (algún programa) que creas que podes executar directamente escribindo o seu nome. **Execútao**.

1.3. Que executable escolliches e en que carpeta (que ruta) está? Puideches abrilo escribindo o seu nome?

1.4. A variable "PATH" é editable. Desde o teu entorno (Windows, Linux), como poderías editala de forma permanente? Hai outras variables de entorno además de PATH? Se é así, escribe unha xunto co seu valor.

🧠 Se queres podes preguntarlle á IA, pero le con calma e asimila a resposta. Usa as túas palabras. A resposta será analizada e cun marxe de confianza, se parece copiada e pegada, non será válida.

1 Se é Windows, desde o inicio rápido (tecla de Windows) escribe cmd.exe. Se é Linux, busca Terminal no menú inicio.

SECCIÓN 2: JAVA E JAVAC

Java é unha especificación definida por Oracle Corporation a través do Java Community Process. Esta especificación inclúe regras sobre como funciona a linguaxe (a sintaxe), como son as librarías estándar e como debe comportarse a Java Virtual Machine.

Un JDK (Java Development Kit) é un paquete software que inclúe o compilador de Java (javac), a JVM e ferramentas.

OpenJDK é unha implementación de código aberto do JDK.

2.1. Que outras implementacións baseadas en OpenJDK existen? (escribe 4 ou 5)

A especificación de Java evoluciona constantemente, polo que hai moitas versións de Java e, consecuentemente, de JDKs.

2.2. Cales son as versións de Java máis importantes? Cales son as principais diferencias que introduciu cada unha?

🧠❓ É posible que algunhas características non as poidas "comprender" completamente polo de agora, pero inténtao o mellor que poidas e elabora unha resposta coas túas palabras, incluso se iso significa poñer *"isto non o entendo ao 100%"*

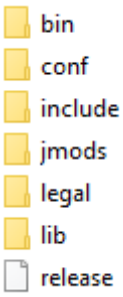
A continuación vamos instalar un JDK co obxectivo de ter as ferramentas java e javac[2] no sistema (java.exe e javac.exe en Windows).

⚠ Se xa instalaches o JDK anteriormente, ou, ao instalar un IDE coma IntelliJ IDEA o JDK foi descargado automáticamente, entón non tes que seguir os seguintes pasos. Comproba se xa tes dispoñible "java" ou "javac" no teu sistema saltando ao punto 2.4 da práctica.

Descarga a versión 26 do OpenJDK desde a web de Oracle:

- https://jdk.java.net/26/

Terás unha carpeta comprimida con estes contidos:



Descompríme a carpeta nunha ubicación adecuada.
En Windows, eses contidos debes metelos nunha nova carpeta **C:\Java\jdk-26**.
En Linux, leva os contidos a unha nova carpeta **/opt/jdk-26.**

2 javac é a abreviación de "Java Compiler"

Agora, **crea** unha nova variable de entorno permanente[3] JAVA_HOME que apunte á túa nova carpeta, C:\Java\jdk-26 ou /opt/jdk-26 en función de se usas Windows ou Linux. Pecha o teu terminal se o tes aberto. Abre un novo terminal e executa:

En Windows, `echo %JAVA_HOME%`
En Linux, `echo $JAVA_HOME`

2.3. Cal é a saída no terminal?

Engade a ruta %JAVA_HOME%\bin á túa variable PATH, de xeito permanente. Desta forma, os executables que están na carpeta "bin" do JDK serán invocables directamente no teu sistema. Despois de facelo, pecha e volve abrir o terminal para ter os cambios.

2.4. Executa nun terminal **java -version** e **javac -version**. Escribe a saída de cada comando.

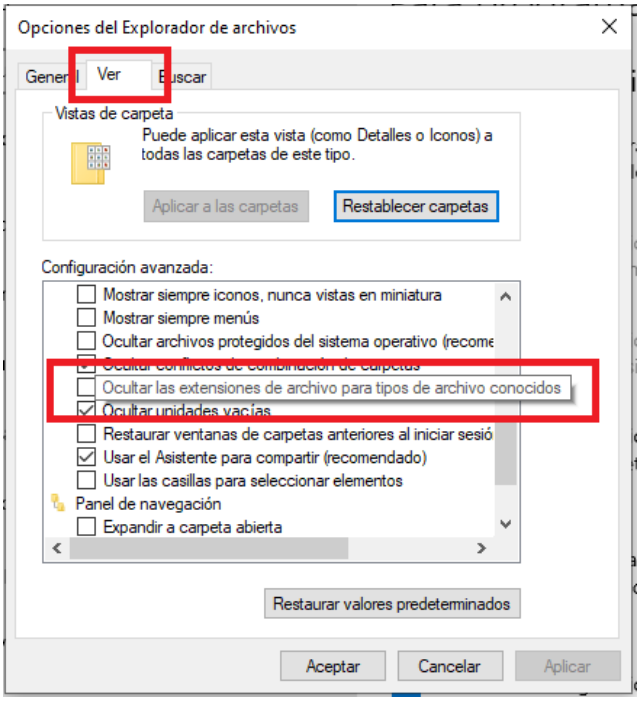
SECCIÓN 3: BREVE CONFIGURACIÓN DO EXPLORADOR DE ARQUIVOS EN WINDOWS

Se usas Windows, o explorador de arquivos por defecto oculta as extensións dos arquivos coñecidos.

É dicir, se tes un ficheiro que se chama MeuArquivo.txt, posiblemente no explorador de arquivos só se mostra "MeuArquivo".

Isto pode producir problemas nos seguintes apartados da práctica, xa que pode facer que escribamos nomes de ficheiros coma "Main.java.txt" e non nos deamos conta.

3.1. Se tes Windows, desde o inicio rápido **abre** Opciones del Explorador de archivos e **desmarca** a opción de "Ocultar extensiones para tipos conocidos".



3 Sabes como facelo? Revisa o apartado 1.4 da práctica

SECCIÓN 4: COMPILANDO E EXECUTANDO PROGRAMAS SINXELOS

Compilar un programa é traducilo a código máquina.

Crea un novo arquivo desde o Bloc de Notas en Windows, nano ou gedit en Linux (ou o que se sexa o teu editor de texto plano sinxelo favorito), e **escribe** o seguinte programa Java:

```
public class Main {
    public static void main(String[] args) {
        int a = 2;
        int b = 3;
        int c = a + b;
        System.out.println(c);
    }
}
```

Despois, **garda** o arquivo co nome “Main.java” nunha ubicación que ti queiras. Aconsellablemente a túa carpeta de usuario.

4.1. Onde gardaches o arquivo no teu ordenador? (Escribe a ruta completa)

Agora vamos a compilar ese programa. Executa **javac Main.java** no teu terminal, desde a mesma ruta onde gardaches o ficheiro.

4.2. Busca un novo arquivo que foi creado automáticamente. Como se chama?

4.3. Ese ficheiro .class, que cres que contén?

- ◉ Código máquina para a arquitectura do teu ordenador
- ○ Bytecode para a JVM

4.4. Que é a máquina virtual de Java (JVM)? Que características ten?

🧠 Podes (se queres) preguntarlle á IA. Resume a información ao teu xeito e escribe aquí o que teñas interiorizado coas túas palabras.

Para inspeccionar a baixo nivel o contido do ficheiro .class executa **javap -c Main**.

4.5. Cal é a saída?

Dalle unha volta con calma á túa resposta e reflexiona ata que punto eres capaz de comprender o que pon 🤔.

Agora, executa o programa escribindo no terminal **java Main**.

4.5. Cal é a saída producida?

Despois, **elimina** o arquivo de código fonte Main.java. Só debe quedar o ficheiro Main.class na túa carpeta.

4.6. É posible executar "java Main" novamente? Que deduces disto?

Vamos a crear un novo programa.

Na túa carpeta, crea e **garda** un novo ficheiro "Echo.java" co seguinte código:

```
import java.util.Scanner;

public class Echo {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter something: ");
        String input = sc.nextLine();

        System.out.println("You typed: " + input);

        sc.close();
    }
}
```

Non o compiles (javac) todavía.

4.7. Proba a executalo con **java Echo**. Cal é a saída?

4.8. Agora sí, compila o programa con **javac Echo.java**. Volve a executalo, e completa a execución escribindo algo. Cal foi a saída completa?

SECCIÓN 5: PROGRAMAS CON VARIOS ARQUIVOS

5.1. Pódense crear programas Java que consistan en máis dun arquivo? Que é un paquete Java? Escribe un exemplo de como se poderían meter dous arquivos .java nun paquete e compilalos. Non omitas ningún detalle que sexa crucial.

🧠 Axúdate da túa IA favorita de xeito responsable, como viñemos facendo ata o de agora. O importante non é dar unha resposta, senón que ti obteñas unha boa base de coñecemento para resolver problemas futuros con soltura e habilidade… Incluso cousas que se lle escapen á IA.

5.2. Se teño dous arquivos Main.java e Helper.java nunha carpeta calqueira, asumindo que a clase Main utiliza a clase Helper… Podo compilalos e executalos dunha vez con javac?

- ◉ Non. Non se pode compilar máis dun ficheiro á vez
- ○ Non. Debo incluilos nun paquete e nunha carpeta src/
- ○ Sí, pero debo obrigatoriamente executar javac Main.java Helper.java
- ○ Sí. Podo, por exemplo, executar javac Main.java

❓ Unha forma de darlle resposta a esta pregunta sería probar esa situación no teu ordenador

Vamos a crear un paquete Java.

Crea unha carpeta “src” na túa carpeta de traballo. Dentro, crea outra carpeta “myapp”.

Crea o arquivo src/myapp/Animal.java e dentro escribe unha clase Animal que conteña unha variable **public static String** co nome **animalFavorito**. O valor desta variable será o teu animal favorito, se tes un. Se non, un calqueira.

A continuación, **crea** o arquivo src/myapp/Main.java e dentro escribe unha clase Main e un método principal **main** que faga un **System.out.println** da variable Animal.animalFavorito.

Non esquezas, en ambos arquivos (Main.java e Animal.java), escribir na primeira liña do arquivo “**package myapp;**”

5.3. Abre un terminal desde a carpeta “src”. Executa **javac myapp/*.java** para compilar o teu código. Cales son os catro arquivos que hai agora na carpeta “myapp”?

5.4. Executa **java myapp.Main**, cal é a saída da execución?

Ter xuntos os arquivos de código fonte (.java) e os arquivos de bytecode (.class) non é unha boa práctica. **Elimina** os ficheiros .class resultado do apartado 5.3.

5.5. Agora, executa **javac -d out myapp/*.java**. Onde están agora os arquivos .class? Explica a xerarquía de directorios.

5.6. Funciona agora a execución java myapp.Main?

- (•) Sí
- ( ) Non

5.7. Que é a variable de entorno CLASSPATH?

🧠 Pregúntalle á IA se queres, ou busca en Internet

5.8. Hai algún xeito de especificarlle ao comando Java onde ten que buscar as clases (sen configurar ningunha variable de entorno)? Tendo o terminal na carpeta "src", que comando lle pasarías á java para que atope os ficheiros .class do teu programa na ubicación que están agora, e así, funcione a execución?

🧠 Podes axudarte da IA, pero lembra verificar todo o que che propoña e probalo no teu ordenador

SECCIÓN 6: EXTRA (NON AVALIABLE)

6.1. Eres capaz de, sen axuda, escribir, compilar e probar un programa que pregunte o teu estado emocional e proporcione distintas respostas en función do que escribas?

6.2. Cal é a diferencia entre un .jar e un .class?

6.3. Como podes xerar un .jar? Como podes executalo?

6.4. Que é Maven? Cal é o ficheiro máis importante dun proxecto con Maven?

6.5. Podes crear un programa Java que sexa capaz de xogar ao pedra-papel-tesoira?

6.6. Que outras ideas para programas sinxelos eres capaz de implementar?