**Cuestiones U1 – *Introducción a la Programación***

* ¿Dónde se ubican los programas para poder ser ejecutados en el ordenador?

**En los discos duros, memoria de almacenamiento.**

* El código **binario** es el código que utilizan internamente los ordenadores.
* Define lenguaje máquina.

**Los lenguajes de programación están divididos en tres grupos principales, los lenguajes maquina, los lenguajes de bajo nivel y los lenguajes de alto nivel.**

**los lenguajes maquina son aquellos en el las instrucciones están escritas en el lenguaje natural del ordenador entendible por él, es decir en binario, hexadecimal octal etc. por lo tanto el microprocesador puede entender directamente las instrucciones y no necesita de una "traducción" por así decirlo.**

* Define algoritmo. Expresa el algoritmo que te permite “*calentar el desayuno en el microondas*” cada mañana.

**Es una secuencia de instrucciones para hacer algo.**

**1º echar leche en el vaso**

**2º mientras que temperatura sea == a frio**

**2.1- meter la leche en el microondas**

**2.2- programar microondas**

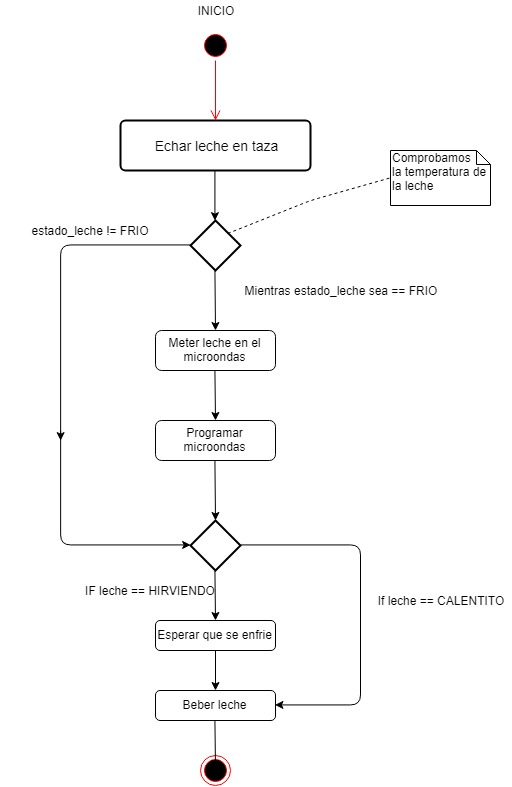
**2.3- cerrar puerta de microondas**

**3º es temperatura == calentito**

**3.1 pasar al paso 5**

**4º esperar a que se enfríe**

**5º beber leche**

****

* ¿Qué es un programa? Un programa está escrito en un **lenguaje de programación**.

**Son secuencias de instrucciones escritas en un lenguaje de programación que permiten a la maquina realizar una actividad.**

* Ventajas de un lenguaje de alto nivel.

**Genera un código mas sencillo y comprensible para el ser humano**

**Generalmente con ellos puedes escribir un código que sea compatible con distintos sistemas operativos y distintos tipos de máquinas.**

**Permite la abstracción y la utilización de paradigmas de la programación como por ejemplo POO.**

* Un programa escrito en un lenguaje de alto nivel, ¿es comprensible directamente por la máquina? ¿Qué se necesita?

**No, necesita compilarse es decir traducir el código que se ha generado en un código comprensible para la máquina.**

* Diferencia entre intérprete y compilador. Ventajas e inconvenientes de uno y otro.

**El interprete a medida que vamos escribiendo va transformando el código esto nos ayuda a que podamos ir detectando los errores a medida que vayamos escribiendo el código (errores en tiempo de compilación), mientras que el compilador traduce el código directamente y luego lo ejecuta. Pues el problema es que se necesita de los dos porque a lo mejor tu escribes el código y es correcto no tiene errores de sintaxis y el interprete te lo da por valido, pero luego cuando compilas el programa cuando ejecutas el programa pueden aparecer errores en tiempo de ejecución.**

**En general, un lenguaje compilado está optimizado para el momento de la ejecución, aunque esto signifique una carga adicional para el programador. Por otro lado, un lenguaje interpretado está optimizado para hacerle la vida más fácil al programador, aunque eso signifique una carga adicional para la máquina.**

* La salida de un compilador es un **ejecutable**. La entrada a un programa enlazador (“linker”) es un .**obj** y la salida es un .**exe**.
* ¿En qué momento se detectan los errores sintácticos en un programa?

**En el momento de la compilación.**

* ¿Qué significa que un programa sea legible? ¿Y reutilizable?

**Legible - claro y fácil de leer**

**Reutilizable – fácil de reutilizar, todo o parte, en otros programas**

* Un estilo, un determinado enfoque a la hora de construir y organizar un programa es un **paradigma** de programación.
* Indica la frase que define la programación procedural. Ídem para la POO.

**“Algoritmos + Estructuras de datos = Programas”**

**“Objetos + Flujo de mensajes = Programas”**

* El **objeto** es el bloque principal de construcción en la POO y tiene **identidad, atributos y comportamientos.**
* Ventajas de la POO.

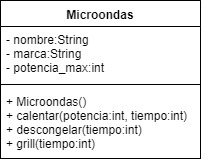
**modela mejor el mundo real, es más intuitivo, describe el problema en términos similares a los que utiliza la mente humana**

**maneja mejor la complejidad del software**

**permite la reutilización obtiene programas más robustos y estables**

**facilita la extensibilidad, la escalabilidad de las aplicaciones**

* Tengo en mi casa un microondas cuya marca es Fagor y tiene una potencia máxima de 1200 w. Es capaz de calentar, descongelar y tiene función de grill. Identifica, para mi objeto Microondas qué serán sus atributos (estado) y sus métodos (comportamiento).



* Explica los conceptos de abstracción y encapsulación.

**Abstracción: consiste en pensar en como son los objetos en la vida real y como se comportan estos. Es decir, sus características como se utilizan las funciones que hacen y eso plasmarlo en el programa.**

**Y la encapsulación seria coger todos los atributos y los métodos y meterlos dentro de una estructura de una clase es decir el hecho de meterlos en las llaves dentro de la clase seria encapsulación**

* ¿Qué se hace en las fases de implementación y pruebas?

**Implementación – (Programación OO - codificación ) – Se traduce el diseño a un lenguaje de programación concreto.**

**Pruebas - se verifica que el producto construido hace lo deseado. Se preparan test con datos de prueba para comprobar el correcto funcionamiento.**

* ¿UML es una metodología de desarrollo de software? Razona la respuesta.

**UML no es una metodología de desarrollo de software OO sino una notación para especificar, construir, visualizar y documentar los elementos de un sistema de software**

* ¿Por qué se dice que Java es compilado e interpretado a la vez?

**Es que Java es un lenguaje que es compilado a un lenguaje intermedio llamado [bytecode](https://en.wikipedia.org/wiki/Java_bytecode" \t "_blank), que después es interpretado. Los creadores de Java querían crear un lenguaje compilado, pero que se pudiera ejecutar en cualquier sistema operativo y procesador sin necesidad de crear varios ejecutables.**

**Es por eso que si quieres ejecutar código Java debes instalar el**[**JRE (Java Runtime Environment)**](https://java.com/es/download/)**, que es el programa que se encarga de interpretar el [bytecode](https://en.wikipedia.org/wiki/Java_bytecode" \t "_blank) al que son compilados los programas de Java.**

**Si deseas compilar código Java no es suficiente instalar el JRE, necesitas el**[**JDK (Java Development Kit)**](http://www.oracle.com/technetwork/java/javase/downloads/index.html)**que incluye el compilador, entre otras herramientas de desarrollo.**

* ¿Qué identifican las extensiones .*java* y .*class*?

**Los programas fuente escritos en java (\*.java ) se compilan (con el compilador java ) para obtener ficheros con extensión . class**

* ¿Cuál es la función de la máquina virtual de Java (JVM)?

**Interpretar Los ficheros \*. class ya que estos no contienen código máquina comprensible por ningún procesador sino que contiene bytecodes , una especie de código de bajo nivel (un lenguaje intermedio)**

* Un fichero .*class*, ¿qué tipo de contenido tiene? ¿es directamente ejecutable?

**Bytecodes Una especie de código de bajo nivel (lenguaje intermedio). No por ello necesitan la JVM**

* Java es de arquitectura neutral y portable, ¿qué significa esto?

**Que permite que sea independiente de la plataforma en la que se ejecuta. Pero se necesita la JVM**

* ¿Qué es un *applet*?

**Es un programa diseñado para ejecutarse dentro de un navegador compatible con Java. La salida es gráfica.**

* ¿Qué es la API de Java?

**Es una colección de componentes software, clases, que el programador puede incluir en sus programas. La API ofrece capacidades de todo tipo: gráficas, matemáticas, componentes de red, etc. Todos estos componentes se agrupan en librerías de clases relacionadas (paquetes).**

* ¿Qué necesitamos instalar en nuestro equipo para poder desarrollar y ejecutar programas Java?

**Para desarrollar un programa Java hay que tener instalado el JDK en el sistema.**

**El JRE y la JVM**

* ¿Qué es *javac*, *javadoc* y *java*?

**javac – compilador a bytecodes**

**javadoc – generador de documentación**

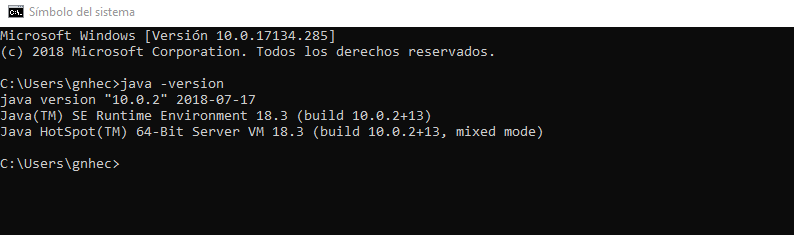
**java – herramienta del JDK para ejecutar programas java invocando a la máquina virtual de java (JVM)**

* Tienes escrito en papel un programa fuente java y has de instalarlo y ejecutarlo en el ordenador. No dispones de ningún entorno de desarrollo que te facilite la tarea de editar, compilar, …. Indica que pasos sigues hasta poder ejecutar el programa.

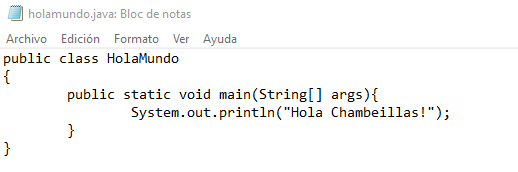
**1º Tienes que instalar el jdk en el ordenador**

<http://www.oracle.com/technetwork/java/javase/downloads/jdk10-downloads-4416644.html>

**para comprobar que tenemos el Java SE Runtime Environment (JRE) instalado en el sistema tenemos que poner java -version**



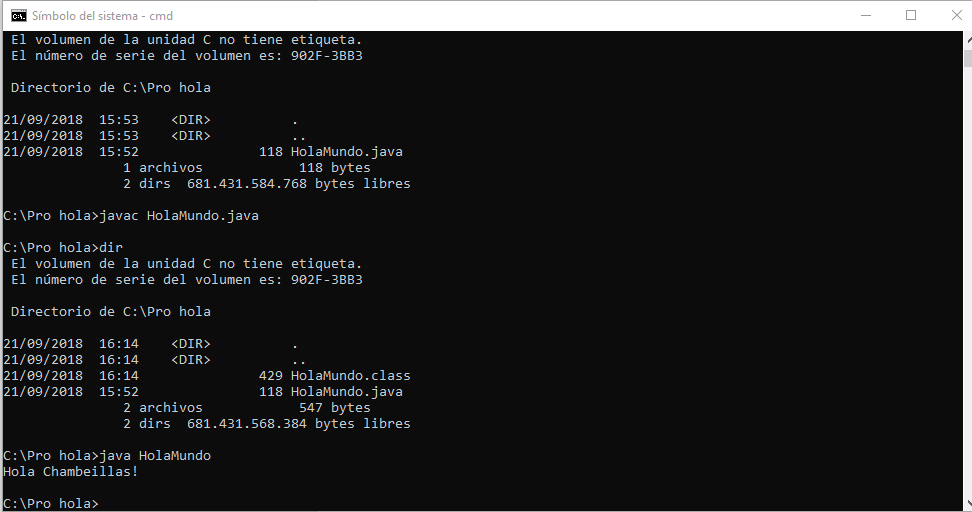
**2º ahora tenemos que escribir en un bloc de notas el programa y guardarlo con la extensión .java**



**3º nos movemos hacia el directorio donde esta guardado el documento que hemos creado con el comando cd y cuando estemos allí escribimos javac y el nombre del documento. (esto sirve para compilarlo)**

**4º entonces se habrá creado el \*.class**

**5º para ejecutarlo solo hay que escribir el java nombreDelArchivo y listo! 😊**



**Sino sale es porque las variables de entorno estan mal y hay que configurar el path**

1. **Hacer clic en el botón derecho de "Equipo".**
2. **Seleccionar "Propiedades".**
3. **Pinchar en "Configuración avanzada del sistema" > "Variables de entorno".**
4. **Seleccionar la variable PATH.**
5. **Pulsar en el botón "Editar".**
6. **En el "Valor de la variable" –sin borrar lo que ya hay– añadir al final un punto y coma ";" seguido de la ruta donde se encuentre el archivo javac.exe. Por ejemplo:**

**Esto es la comprobación**

