

PAC 1. UF4.

*Introducción a la
Programación Orientada a
Objetos.*

1. ¿Cuál es la principal característica de la programación orientada a objetos? Explica qué es y sus diferentes mecanismos.
2. ¿Qué es una clase? ¿Qué son los atributos y los métodos?
3. ¿Qué es la palabra reservada this? ¿A qué hace referencia?
4. ¿Qué es Java?
5. Define las siguientes palabras reservadas:
 - Final.
 - Static.
 - Import.
 - Instanceof.
6. ¿Qué tipo de datos tenemos en Java? Indica sus características.
7. ¿Qué es una constante? Como lo definimos en Java?
8. ¿Cómo se realiza el paso de parámetros en Java? Explica el paso de parámetros a funciones.
9. Rellena la tabla con Si o NO dependiendo de la visibilidad que indica el modificador.

Modificador	Clase	Subclase	Package	Otro
<i>Public</i>				
<i>Protected</i>				
<i>Private</i>				
<i>No especificado</i>				

10. Realiza la instalación del IDE Eclipse. Crea un programa en Eclipse con una función main que pida 2 números por teclado y muestre:
 - a. El número mayor.
 - b. El factorial del primer número.

1. ¿Cuál es la principal característica de la programación orientada a objetos? Explica qué es y sus diferentes mecanismos.

En la programación orientada a objetos una de las principales características es la abstracción. Permite encapsular el código, aislarlo del propio diseño, por lo que cuando utilizamos el software, sabemos como utilizarlo, pero no el código escrito para usarlo. Los mecanismos que utilizaremos en la abstracción son, **los procedimientos y las funciones**, que se ubicarán junto al programa principal que los utiliza, **los módulos**, parecidos a las funciones o procedimientos pero podremos ubicarlos en zonas no accesibles privadas.

Además tenemos los **datos de tipo abstracto** que son aquellos definidos por el programador que incluyen un rango de valor así como métodos para la manipulación de esa información.

Y por último tenemos los **objetos** al que se añaden características como los métodos, herencia o poliformismo.

2. ¿Qué es una clase? ¿Qué son los atributos y los métodos?

Una clase es la forma en como definimos a un objeto. Es el elemento que describe los componentes de un objeto de generalmente. A dicho objeto cuando le damos una serie de valores y usamos esos componentes entonces creamos una clase.

Los atributos o propiedades hacen referencia a una característica de un objeto. Un atributo de clase se establece similarmente a como declaramos una variable.

Los métodos son aquellos que determinan las operaciones que pueden realizarse sobre los atributos, se parecen a las funciones pero se dedican a modificar las propiedades de una clase. Dan funcionalidad a la clase.

3. ¿Qué es la palabra reservada this? ¿A qué hace referencia?

This es una palabra reservada que se utiliza cuando queremos hacer referencia al objeto que estamos programando. Cuando es utilizada delante de un atributo o método, estamos indicando que ese es el elemento que trataremos. Con **this** podemos pasar parámetros con el mismo nombre que los atributos de la clase, ya que el compilador distinguirá cual es el atributo y cual es el parámetro.

4. ¿Qué es Java?

Java es un lenguaje de programación y plataforma informática comercializada en 1995 por Sun Microsystems. Existen multitud de aplicaciones y sitios web que no funcionarán a menos que se tenga instalado Java. Java es rápido, seguro y fiable.

Es un lenguaje interpretado, el código que diseña se denomina bytecode y, se puede interpretar a través de una maquina virtual.

Es un lenguaje orientado a objetos, y multiplataforma, su compilador produce un código binario de tipo universal, se puede ejecutar en cualquier tipo de maquina virtual que admita la versión utilizada.

Java cuenta con una gran colección de clases agrupadas en diferentes directorios. Estas clases sirven al usuario para realizar alguna tarea determinada sin necesidad de tenerla que implementar.

5. Define las siguientes palabras reservadas:

- **Final.** Esta palabra reservada indica que a una variable solo se le puede asignar un valor u objeto una única vez. Por ejemplo en una aplicación donde definamos algún valor que permanezca constante durante la ejecución, y definiéndolo como variable, perderíamos el control cambiando algún valor, por lo que agregando este modificador únicamente podremos asignarle un único valor.
- **Static.** Una clase, método o campo declarado como static puede ser accedido sin tener que instancia a u objeto de la clase utilizando esta palabra reservada.
- **Import.** Con esta palabra importaremos clases de un paquete. Podemos importar una clase individualmente o importar clases declaradas públicas de un paquete completo.
- **Void.** Indica que algo está vacío, que no tiene tipo. Si un método no devuelve ningún resultado o valor se indica que devuelve void (vacío).
- **Instanceof.** Utilizamos esta palabra para conocer si un objeto es de un tipo determinado (clase, interfaz o array).

6. ¿Qué tipo de datos tenemos en Java? Indica sus características.

Tenemos dos tipos de datos:

- **Tipos primitivos.** No son objetos y no necesitan invocación para ser creados. Aquí tenemos:
 - **Byte.** Tipo entero de 1 byte y rango -128 a 127.
 - **Short.** Tipo entero de 2 byte y rango -32768 a 32767.
 - **Int.** Tipo entero de 4 byte y rango 2×10^9 .
 - **Long.** Tipo entero de 8 byte y rango muy grande.
 - **Float.** Tipo decimal simple de 4 byte y rango muy grande.
 - **Double.** Tipo decimal doble de 8 byte y rango muy grande.
 - **Char.** Tipo carácter simple de 2 byte.
 - **Boolean.** Tipo valor verdadero o falso de 1 byte..
- **Tipos objeto.** Necesitan invocación para ser creados. Tenemos:
 - Tipos estándar de Java. Donde tenemos:
 1. String.
 2. Otros tipos.
 - Tipos definidos por el usuario. Cualquier método que haga el programador.
 - Arrays. Elementos para formar una matriz. Lo consideramos objeto especial que no tiene métodos.

- Tipos wrapper. Equivalentes a los primitivos pero como objetos.

7. ¿Qué es una constante? Como lo definimos en Java?

Una constante es todo valor que se mantiene igual durante su ejecución. Utilizamos la palabra final para declarar constantes y debe ser inicializada en el momento que se crea y no podrá ser modificada a lo largo de toda la ejecución del programa.

8. ¿Cómo se realiza el paso de parámetros en Java? Explica el paso de parámetros a funciones.

En Java siempre pasamos los parámetros por valor. Es decir, cuando un argumento se pasa a una función, la función recibe una copia del valor original. Por lo tanto si la función modifica el parámetro, sólo la copia cambia y el valor original permanece intacto.

Una función con parámetros necesita cierto tipo de información para poder realizarse. Esta puede recibir tantos parámetros como sea necesario. Puede recibir valores simples, numéricos, booleanos, o strings. Cuando la función recibe un parámetro, éste funciona como si se tratara de una variable disponible para la función inicializada con el valor que se le pasa a la función.

9. Rellena la tabla con Si o NO dependiendo de la visibilidad que indica el modificador.

Modificador	Clase	Subclase	Package	Otro
<i>Public</i>	Sí	Sí	Sí	Sí
<i>Protected</i>	Sí	Sí	Sí	No
<i>Private</i>	Sí	No	No	No
<i>No especificado</i>	Sí	Sí	No	No

10. Realiza la instalación del IDE Eclipse. Crea un programa en Eclipse con una función main que pida 2 números por teclado y muestre:

- El número mayor.
- El factorial del primer número.

Se adjunta en el archivo comprimido el proyecto completo de Eclipse junto con el programa. El código contenido es el siguiente:

```
import java.util.Scanner;
public class UF4_PAC1 {

    public static void main (String [] args) {

        Scanner sc= new Scanner(System.in);
        int num1, num2;
        int factorial=1;
```

```
System.out.print("Introduce numero 1: ");  
num1 = sc.nextInt();
```

```
int aux=num1;
```

```
System.out.print("Introduce numero 2: ");  
num2 = sc.nextInt();
```

```
if (num1 > num2) {System.out.println("El numero mayor es: " + num1);}  
else {System.out.println("El numero mayor es:" + num2);}
```

```
while (aux!=0) {  
    factorial=factorial*aux;  
    aux--;  
}
```

```
System.out.println("El factorial del numero 1 es: " + factorial );
```

```
}
```

```
}
```