

## Trayecto Programador

### Glosario

- **Clase:** La clase es una abstracción que agrupa elementos a partir de atributos que definen los elementos.
- **Atributos:** Son características específicas de una clase es donde se almacena la información de un objeto.
- **Objeto:** Elemento que hace parte de la clase.
- **Instancia:** Dar valores a los atributos de la clase para definir un objeto.
- **Java:** Es un lenguaje de programación orientada a objetos.
- **Máquina virtual:** Es la base donde se desarrolla el programa, la máquina virtual de java es el JDK.
- **Composición:** Relación fuerte, quiere decir que una clase que está compuesta por otra no podría existir sin dicha clase.
- **Agrupación:** Relación débil, quiere decir que la clase agrupada no necesariamente deja de existir si la otra deja de existir.
- **Constructor:** Método que permite crear la instancia de un objeto.
- **Constructor sobrecargado:**  
Es un constructor que está definido más de una vez.
- **Parámetro:** Un valor que viene como entrada para definir un proceso.

- **This:** Acceso.
- **Método main:** Método principal del programa.
- **New:** Crear una nueva instancia.
- **Tipo de dato:** Espacio de memoria que se le pide a la máquina para almacenar los datos.
- **Compilador:** Es un programa informático que traduce un programa escrito en un lenguaje de programación a otro usualmente al lenguaje de máquina.
- **Class:** Contiene el bytecode el cual no entienden los humanos, pero si la máquina virtual de java.
- **Contrato:** Es donde se estipulan los procesos de entrada, proceso y salida de cada método.
- **Polimorfismo:** Se da en Métodos que requieren de comportamiento diferente en cada clase específica.
- **Clase abstracta:** Es aquella clase que por lo menos tiene un método abstracto (la clase no se podrá instanciar).
- **Método abstracto:** Es un método que está declarado mas no desarrollado.
- **Método get:** Sirve para entregar el valor de los atributos encapsulados.
- **Método set:** Asignan valores a atributos encapsulados en una clase.

- **Documentación:** Se utiliza con el fin de agregar detalles importantes como el autor del programa, la fecha de desarrollo, etc, al inicio de la clase.
- **Comentarios:** Se utilizan los comentarios para explicar detalles sobre el desarrollo del programa en la misma estructura del código como tal.
- **Null:** Por ejemplo, si un objeto fue declarado, pero no se instanció, existe un valor especial llamado “null” que se usa para esta situación, el null nos indica que un objeto no se ha instanciado.
- **Palabras reservadas:** Java cuenta con palabras reservadas que tienen un uso específico, y las cuales no pueden ser usadas para crear variables
- **Acceso privado:** Los elementos solo se pueden utilizar dentro de la clase que los define.
- **Acceso protegido:** Los elementos sólo se puede utilizar dentro de la clase que los define o desde una subclase definida en el mismo paquete.
- **Acceso public:** Los elementos se pueden utilizar libremente.
- **Encapsulamiento:** Indica que los atributos de la clase están encapsulados, es decir, aquellos que tienen visibilidad private protected.
- **Variables:** utilizamos variables para almacenar valores, para realizar cálculos, almacenar información, etc. estas variables se guardan en un espacio específico en la memoria.

Existen distintas clases de variables, como int (enteras), float (decimales), char (caracter), etc.

- **Long:** Espacio de almacenamiento expandido para números enteros.
- Clase abstracta no lleva constructor sobrecargado.
- No es necesario que la clase sea abstracta para que sea super clase.
- La composición sobre otra clase se invoca mediante un atributo de la clase que se quiere invocar, no olvidar poner los métodos get y set.
- **Cast de objetos:** No es obligatoria la clase abstracta, si es obligatoria la jerarquía, el cast se utiliza para recuperar datos dentro de una jerarquía.