**TRABAJO PRÁCTICO 1**

1. Es un *“Tipo Abstracto de Datos”*.
2. Se produce en dos niveles, a nivel de clase y a nivel de paquete.
3. - **Semejanzas:** Son bloques de código que permiten la reutilización de código.

* **Diferencias:**

\_ *Retorno de valor:* Las funciones devuelven un valor, los procedimientos no y los métodos son similares a las funciones pero estás asociados a una clase.

\_ *Invocación:* Las funciones se invocan utilizando su nombre y, a menudo, se utilizan para calcular y devolver un valor, los procedimientos se invocan para realizar una tarea sin necesidad de retorno de valor y pueden modificar los datos directamente o realizar operaciones sin devolver un resultado explícito, y los métodos se invocan utilizando una instancia de la clase a la que pertenecen y pueden acceder a los datos interno de esa instancia.

\_ *Asociación:* Las funciones y los procedimientos son independientes y pueden existir por si mismos, mientras que los métodos están asociados a una clase y pueden acceder a los atributos y métodos de esa clase.

1. UML significa *“Unified Modeling Lenguage”* (Lenguaje Unificado de Modelado) y es un lenguaje de modelado estándar que se utiliza para visualizar, especificar, construir y documentar los artefactos de un sistema de software.

En UML, una clase se representa con un rectángulo dividido en tres secciones:

1. **Nombre de la Clase:** Se coloca en la parte superior del rectángulo.
2. **Atributos:** Se colocan en la sección media y se muestra en el formato *“nombre: tipo de dato”*.
3. **Métodos:** Se colocan en la parte inferior y se muestran en el formato *“nombre(argumentos): tipo de retorno”*.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | No se aplica a clases | Solo se aplica a atributos | Solo se aplica a clases | Se aplica a atributos, métodos y clases |
| Public |  |  |  | X |
| Private | X |  |  |  |
| Protected | X |  |  |  |
| Static | X |  |  |  |
| Final |  |  |  | X |
| Primera letra en minúscula | X | X |  |  |
| Primera letra en mayúscula |  |  | X |  |

1. Un constructor:
   * FALSO
   * VERDADERO
   * FALSO
   * FALSO
   * VERDADERO
   * FALSO
   * VERDADERO
   * FALSO
   * FALSO
   * VERDADERO

Un método:

* + FALSO
  + VERDADERO
  + VERDADERO
  + VERDADERO
  + FALSO
  + FALSO

1. Se necesita que cualquiera pueda acceder al color de un vehículo. Entonces, declaro color como: **“public”**
2. Se necesita que color se pueda acceder a través no sólo de vehículo, sí no ahora también de Buses, y como todos sabemos un bus es un tipo de vehículo, entonces también deberá tener acceso a color. Entonces, declaro color como: **“protected”**
3. Se necesita que color se pueda acceder solamente para vehículo. Entonces, declaro color como: **“private”**