Desarrollo de Aplicaciones Distribuidas

Introducción

Para empezar miraremos algunos conceptos básicos de la programación orientada a objetos.

Objetos

Un objeto es un elemento con diferentes atributos, un ejemplo sería el objeto Persona, que tendría los atributos DNI, Nombre y Apellidos.

Métotos

Son funcionalidades aplicacas a cada objeto

Clases

Son implementaciones totales o parciales de un tipo abstracto de datos. Son la representación de objetos del mundo real

Aspectos a tener en cuenta

- El código debe ser reutilizable, de forma que un mismo código se pueda utilizar en diferentes programas.
- El código también debe ser extensible, de forma que podamos añadir funcionalidad a este sin que afecte a aplicaciones anteriores

Objetos VS Invocación de Funciones

Un mensaje es una forma de comunicación entre objetos mientras que la invocación de una función o método será sobre un objeto.

Herencia

Una clase puede heredar atributos y métodos de otra clase, esto nos permite:

- Reciclar código
- Nuevos Comportamientos
- Cambio de comportamiento
- Jerarquía de clases

Aspectos a Tener en Cuenta en Herencia

- No Hay herencia múltiple
- Se usa la plabra clave extends
- Toda clase hereda implícitamente de object
- Solo se heredan clases públicas y protegidas
- Para acceder a métodos del padre se usa Super

Polimorfismos

CONSTRUCCIÓN

JTable

La JTable está asociada a AbstractTabletModel. No podemos crear un objeto de esta clase ya que es una clase abstracta.

From:
https://www.knoppia.net/ - **Knoppia**

Permanent link:
https://www.knoppia.net/doku.php?id=dad&rev=1695641117

Last update: **2023/09/25 11:25**