**Conceptos**

**Herencia**

* La herencia en POO es un mecanismo que nos permite extender de otra clase Para la reutilización de código y facilidad de compartir métodos y datos entre clases, subclases y objetos.

**Polimorfismo**

* El polimorfismo en programación orientada a objetos es la capacidad de un objeto para tomar diferentes formas según sea especificado. Va muy ligado con la herencia.

**Herencia múltiple:**

* Es la característica que nos permite heredar o utilizar comportamientos y características de una superclase.

**Herencia múltiple de interfaz:**

* Este concepto es manejado en java dado que no nos permite utilizar la herencia múltiple como en otros lenguajes. por ello se utilizan la interfaz la cual nos permite implementar el mismo manejo a través de otra sintaxis y una vez implementada podremos utilizar los comportamientos y características de esta interfaz

**Tipos de Polimorfismos**

**Polimorfismo (sobrecarga):**

* Este tipo de polimorfismo nos enseña como el lenguaje tiene la capacidad de tomar la selección entre dos tipos de métodos con parámetros diferentes y al recibir sus datos creando un objeto, en tiempo de compilación escoge cual es el indicado para dicho proceso

**Polimorfismo paramétrico:**

* Este polimorfismo se ve implicado en el momento de tener un método el cual podría tomar diferentes acciones sin necesidad de especificar su tipo (genéricos)

**Polimorfismo de inclusión:**

* Esta clase de polimorfismo muestra la manera en la cual el método puede tomar diferente forma a la hora de heredar. Podríamos dar como ejemplo cuando el método en la superclase es abstracto, pero en la siguiente clase al utilizarlo ya no lo es. Esta capacidad de poder reutilizar y cambiar su forma se le llama polimorfismo de inclusión

**Polimorfismo de interfaz:**

* Permite que tratemos a los objetos como instancias de interfaz. La interfaz tiene métodos en los cuales se especifica que debe hacer mas no le damos la funcionalidad del cómo hacerlo. Se deben definir si o si los métodos de la interfaz en la clase que la implemente.