Notas

Polimorfismo

El **polimorfismo** es la capacidad de un objeto de adoptar múltiples formas.

Esto permite que el mismo método tenga diferentes comportamientos según el tipo real del objeto en tiempo de ejecución.

Una referencia de la **superclase** puede apuntar a un objeto de cualquier **subclase**.

Upcasting y Downcasting

Persona p = **new Estudiante**(); // upcasting implícito

Estudiante e = (Estudiante) p; // downcasting explícito

Arreglos y polimorfismo

Upcasting y Downcasting

Persona p = **new Estudiante**(); // upcasting implícito

Estudiante e = (Estudiante) p; // downcasting explícito

- **Upcasting:** seguro, siempre válido.
- **Downcasting:** puede lanzar ClassCastException si la referencia no es del tipo esperado.

Arreglos y polimorfismo

Una de las ventajas del polimorfismo es poder trabajar con colecciones de la superclase y almacenar objetos de distintas subclases. Esto facilita el manejo uniforme de diferentes tipos de objetos.

Clases abstractas y herencia

Arreglos y polimorfismo

Una de las ventajas del polimorfismo es poder trabajar con colecciones de la superclase y almacenar objetos de distintas subclases. Esto facilita el manejo uniforme de diferentes tipos de objetos.

```
import java.util.ArrayList;

public class Main {

public static void main(String[] args) {

ArrayList<Persona> personas = new ArrayList<>();

personas.add(new Estudiante());

personas.add(new Profesor());
```