

POO con Java

Guía de Ejercicios Prácticos

Ing. Luisina de Paula



```
error_mod.use_x = False
error_mod.use_y = True
error_mod.use_z = False
operation = "MIRROR_Z"
error_mod.use_x = False
error_mod.use_y = False
error_mod.use_z = True

selection at the end -add
obj.select= 1
obj.select=1
context.scene.objects.active
("Selected" + str(modifier
error_ob.select = 0
bpy.context.selected_obj
data.objects[one.name].select
print("please select exactly
OPERATOR CLASSES -----
```

TODO { ODE </>

www.todocodeacademy.com

Ejercicio Integrador: Clases Abstractas e Interfaces

Un fanático de Pokémon desea implementar para el modelado de un videojuego los diferentes ataques de cada una de estas criaturas. Para ello, cuenta con una clase abstracta llamada `Pokemon`, la cual posee los atributos: `num_pokedex`, `nombrePokemon`, `pesoPokemon`, `sexo`, `temporadaQueAparece` y `tipo`, e implementa métodos para los ataques comunes que suele tener la mayoría, entre ellos se encuentran: `atacarPlacaje()`, `atacarArañazo()` y `atacarMordisco()`. Sin embargo, este fanático también desarrolló una serie de interfaces para contemplar los ataques de Pokémons de cierto tipo:

- **IElectrico:** con los métodos `atacarImpactrueno()`, `atacarPunioTrueno()`, `atacarRayo()`, `atacarRayoCarga()`.
- **IPlanta:** con los métodos `atacarParalizar()`, `atacarDrenaje()`, `atacarHojaAfilada()`, `atacarLatigoCepa()`.
- **IFuego:** con los métodos `atacarPunioFuego()`, `atacarAscuas()`, `atacarLanzallamas()`.
- **IAgua:** con los métodos `atacarHidrobomba()`, `atacarPistolaAgua()`, `atacarBurbuja()`, `atacarHidropulso()`.

A partir de estas interfaces, el Pokefanático desea crear las clases que manejen a los personajes principales del videojuego, los cuales son los pokemons starters de la primera temporada (Charmander, Bulbasaur y Squirtle) y Pikachu; para ello tener en cuenta que: Charmander es de tipo fuego, Bulbasaur es de tipo planta, Squirtle es de tipo agua y Pikachu de tipo eléctrico.

Una vez implementadas la clase abstracta e interfaces, sobrescribir los métodos correspondientes para adaptarlos a cada Pokémon mostrando un mensaje en pantalla que indique qué Pokémon es y qué ataque está realizando, por ejemplo: "Soy Charmander y estoy atacando con Ascuas" o "Soy Pikachu y estoy atacando con placaje". Luego de realizar lo mencionado, crear las instancias necesarias y llamar a cada uno de los métodos de cada `Pokemon`.