

Taller 5 Python

Pablo Bernal García

1025888826

Grupo 2

Taller 5 Python – Respuestas

1. Se debe sobrescribir el método ruido() en la clase Pajaro:

```
def ruido(self):
```

```
    print("cantar y silbar")
```

2. Si se crea la clase Pez sin definir nada nuevo, va a heredar el constructor, métodos y atributos accesibles de su clase padre Animal.
3. Los atributos nombre y edad de la clase SerVivo son heredados por la clase Animal. Para que los atributos sean inicializados correctamente en Animal, se debe llamar al constructor de la clase SerVivo usando super():

```
class Animal(SerVivo):
```

```
    def __init__(self, nombre, edad, raza):
```

```
        super().__init__(nombre, edad)
```

```
        self._raza = raza
```

```
        Animal._totalCreados += 1
```

4. Se puede usar el super():

```
class Animal(SerVivo):  
  
    def setNombre(self, nombre):  
  
        super().setNombre(nombre)  
  
    def getNombre(self):  
  
        return super().getNombre()
```

5. El atributo totalCreados es heredado por las demás clases, pero al definirlo de nuevo en las clases hijas, se ocultan los atributos.

6. Los métodos getTotalCreados en las clases hijas están sobrescribiendo el método del padre.

7. Una clase que herede de Persona va a heredar todos sus métodos y también los demás métodos que Persona heredó de SerVivo

8. El método aduenarAnimal puede recibir cualquier objeto que tenga el método ruido, por lo que se le podría pasar un objeto de la clase Animal o cualquiera de sus subclases.

9. Se debe llamar al método de clase de SerVivo getTotalCreados():

```
totalSeresVivos = SerVivo.getTotalCreados()
```

10. El método getRaza en Perro sobrecargaría al método getRaza de Animal, puesto que su firma es diferente al recibir un parámetro extra, el cual es “tipo”.