|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **INSTRUMENTOS DE EVALUACIÓN** | | |  |
| Código: FO-FIT-129 | Versión: 001 | Página 1 de 1 |

|  |  |
| --- | --- |
| **Instrumento de Evaluación:** | Taller de POO |
| **Nombre de Estudiante:** | DORIS CARMENZA GIRLADO MEJIA |
| **Guía de Aprendizaje:** | Introducción a POO II |
| **Fecha:** | 29/02/204 |
| **Jornada:** |  |
| **Nombre del Docente:** | James Mosquera Rentería |

**Descripción**: En este instrumento, los estudiantes deben desarrollar un ejercicio práctico que aplique los conceptos de la POO en Python.

**Anuncio**:

Construye dos superclases, donde una será la clase empleados que pida en consola el nombre del empleado en un método de instancia.

class Persona:  
 def \_\_init\_\_(self, nombre):  
 self.nombre = nombre  
  
 def saludar(self):  
 print(f"Hola, mi nombre es {self.nombre}")  
  
class Empleado(Persona):  
 def \_\_init\_\_(self, nombre, cargo, departamento):  
 super().\_\_init\_\_(nombre)  
 self.cargo = cargo  
 self.departamento = departamento  
  
 def pedir\_nombre(self):  
 self.nombre = input("Ingrese su nombre: ")  
  
 def mostrar\_informacion(self):  
 print(f"Nombre: {self.nombre}")  
 print(f"Cargo: {self.cargo}")  
 print(f"Departamento: {self.departamento}")

La segunda clase será salario que pida en consola el salario del empleado en un método de instancia.

class Salario:  
 def \_\_init\_\_(self, nombre, cargo, departamento, salario):  
 super().\_\_init\_\_(nombre)  
 self.cargo = cargo  
 self.departamento = departamento  
 self.salario = salario  
  
 def pedir\_salario(self):  
 try:  
 self.salario = float(input("Ingrese su salario: "))  
 except ValueError:  
 print("Error: El salario debe ser un número.")  
  
 def mostrar\_informacion(self):  
 print(f"Nombre: {self.nombre}")  
 print(f"Cargo: {self.cargo}")  
 print(f"Departamento: {self.departamento}")  
 print(f"Salario: {self.salario}")

Cree una clase hija llamada Designación que herede las dos clases anteriores y que tenga un método de instancia que designe el cargo del empleado.

class Designacion:  
 def \_\_init\_\_(self, nombre, cargo, departamento, salario):  
 super().\_\_init\_\_(nombre, cargo, departamento, salario)  
  
 def designar\_cargo(self, nuevo\_cargo):  
 self.cargo = nuevo\_cargo  
 print(f"El cargo del empleado {self.nombre} ha sido actualizado a {self.cargo}")

Verifique el código, instanciando un objeto de la clase hija, verificando si el objeto tiene el método nombre y salario.

**Entrega**:

**Enlace de con el código del ejercicio**

|  |  |  |  |
| --- | --- | --- | --- |
| **N.º** | **Aspecto para evaluar** | **Si** | **No** |
| 1 | Creación de Clases: El estudiante debe crear las clases en Python. |  |  |
| 2 | Definición de Atributos: Los atributos de clase deben ser definidos correctamente. |  |  |
| 3 | Aplicación de Conceptos de POO: aplicación de conceptos como encapsulamiento, herencia, y métodos de clases. |  |  |

**Firma del estudiante Firma del instructor**