

_5class_poo/class_empleado.py

```
1 # Definición de la clase base Persona
2 class Persona:
3     # Atributo de clase
4     contador_personas = 0
5
6     # Método constructor (__init__) que inicializa los atributos de instancia
7     def __init__(self, nombre, edad):
8         self.nombre = nombre
9         self.edad = edad
10        Persona.contador_personas += 1 # Incrementa el contador de personas
11
12    # Método de instancia
13    def saludar(self):
14        print(f"Hola, mi nombre es {self.nombre} y tengo {self.edad} años.")
15
16    # Método de clase
17    @classmethod
18    def obtener_numero_personas(cls):
19        return cls.contador_personas
20
21    # Método estático
22    @staticmethod
23    def descripcion():
24        return "La clase Persona representa a un ser humano."
25
26    # Método especial para representación de cadena
27    def __str__(self):
28        return f"{self.nombre}, {self.edad} años"
29
30    # Método especial para representación oficial
31    def __repr__(self):
32        return f"Persona(nombre={self.nombre}, edad={self.edad})"
33
34 # Definición de la subclase Empleado que hereda de Persona
35 class Empleado(Persona):
36     # Método constructor que inicializa atributos adicionales
37     def __init__(self, nombre, edad, salario):
38         super().__init__(nombre, edad) # Llama al constructor de la clase base
39         self.salario = salario
40
41    # Método de instancia adicional
42    def mostrar_informacion(self):
43        self.saludar()
44        print(f"Mi salario es {self.salario}.")
45
46    # Método especial para representación de cadena
47    def __str__(self):
48        return f"{self.nombre}, {self.edad} años, salario: {self.salario}"
49
50    # Método especial para representación oficial
51    def __repr__(self):
```

```
52         return f"Empleado(nombre={self.nombre}, edad={self.edad}, salario={self.salarior})"
53
54
55 # Definición de la clase Matematica con un método estático
56 class Matematica:
57     @staticmethod
58     def sumar(a, b):
59         return a + b
60
61 # Ejemplos de uso
62
63 # Creación de instancias de la clase Persona
64 persona1 = Persona("Juan", 30)
65 persona2 = Persona("Ana", 25)
66
67 # Llamada a métodos de instancia
68 persona1.saludar() # Salida: Hola, mi nombre es Juan y tengo 30 años.
69 persona2.saludar() # Salida: Hola, mi nombre es Ana y tengo 25 años.
70
71 # Llamada a método de clase
72 print(Persona.obtener_numero_personas()) # Salida: 2
73
74 # Llamada a método estático
75 print(Persona.descripcion()) # Salida: La clase Persona representa a un ser humano.
76
77 # Creación de instancia de la subclase Empleado
78 empleado1 = Empleado("Luis", 28, 50000)
79
80 # Llamada a métodos de instancia de Empleado
81 empleado1.mostrar_informacion()
82 # Salida:
83 # Hola, mi nombre es Luis y tengo 28 años.
84 # Mi salario es 50000.
85
86 # Llamada a método estático de Matematica
87 print(Matematica.sumar(5, 7)) # Salida: 12
88
89 # Uso de métodos especiales para representación de cadena
90 print(persona1) # Salida: Juan, 30 años
91 print(repr(persona1)) # Salida: Persona(nombre=Juan, edad=30)
92 print(empleado1) # Salida: Luis, 28 años, salario: 50000
93 print(repr(empleado1)) # Salida: Empleado(nombre=Luis, edad=28, salario=50000)
94
95
96
97
98
```