



PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

PROGRAMACION ORIENTADA A OBJETOS

Presentado a: Instructor César Marín Cuéllar Chacón
Por Aprendiz: Cristian David Yalandra
Ficha: 3312932
Competencia: POO

Tecnólogo en Análisis y Desarrollo de Software
Servicio Nacional de Aprendizaje SENA
Centro de Teleinformática y Producción Industrial
Regional Cauca

Popayán, día **25 de 02** del año **2026**



PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

Tabla de Contenido

Contenido

1.	Actividad	3
	Segundo Punto: Realizar la actividad didáctica del objeto virtual y anexar el resultado:	3
2.	Actividad 4	3
	Enunciado	3
	Respuesta	4



PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

1. Actividad

Segundo Punto: Realizar la actividad didáctica del objeto virtual y anexar el resultado:



Conceptos básicos programación orientada a objetos



¡Excelente! Has superado la actividad.



- 1 Permite distinguir un objeto de otro.
- 2 Representa las características del objeto. Son los parámetros que lo definen y lo diferencian de objetos del mismo tipo.
- 3 Representan el comportamiento del objeto. Acciones que realizan o manejan los datos.



Finalizar



2. Actividad 4

Enunciado

Muestre con un ejemplo o con varios ejemplos como puedo representar cada uno de los ítems en el lenguaje de programación python ¿Cómo se crea una clase?

¿Cómo se crea el constructor de una clase?

¿Cómo se inicializa un atributo de una clase?

¿Cómo se crea un método o función en una clase?

¿Cómo se crea un objeto de una clase?



PROCESO DE GESTIÓN DE LA FORMACIÓN PROFESIONAL INTEGRAL FORMATO ENTREGA DE EVIDENCIAS

¿Cómo modiflico un atributo de un objeto?

¿Cómo accedo a un método de la clase?

Respuesta

```
➊ ejerClass1.py > ...
1   #crear una clase
2   class Persona:
3       pass
4   #crear un contructor de una clase
5   class Persona:
6       def __init__(self):
7           print("Se ha creado una persona")
8   #inicializar un atributo de una clase
9   class Persona:
10      def __init__(self, nombre, edad):
11          self.nombre = nombre
12          self.edad = edad
13   #crear un metodo o funcion de una clase
14   class Persona:
15       def __init__(self, nombre, edad):
16           self.nombre = nombre
17           self.edad = edad
18
19       def saludar(self):
20           print("Hola, mi nombre es", self.nombre)
21
22       def cumplir_anios(self):
23           self.edad += 1
24   #crear objeto de una clase
25
26 persona1 = Persona("Carlos", 20)
27 persona2 = Persona("Ana", 25)
28
29 #modificar un atributo de un objeto
30
31 persona1.edad = 21
32 persona1.cumplir_anios()
33
34 #acceder a metodo de clase
35
36 persona1.saludar()
```