

Untitled

#Practica 4. Herencia # 1. Crear una clase Ticket con los siguientes atributos: id, tipo (por ejemplo: software, prueba), prioridad (alta, media, baja) y estado (por defecto "pendiente")

#2. Crear dos tickets de ejemplo y mostrar en pantalla

```
class Ticket: def init(self, id, tipo, prioridad ): self.id = id self.tipo = tipo self.prioridad = prioridad
```

```
def mostrar_id(self):  
    return f"El id del ticket es: {self.id}"  
def mostrar_tipo(self):  
    return f"El tipo del ticket es: {self.tipo}"  
def mostrar_prioridad(self):  
    return f"La prioridad del ticket es: {self.prioridad}"  
def mostrar_estado(self):  
    return f"El estado del ticket es: pendiente"
```

```
ticket1 = Ticket(1, "software", "alta") print(ticket1.mostrar_id())  
print(ticket1.mostrar_tipo()) print(ticket1.mostrar_prioridad())  
print(ticket1.mostrar_estado())
```

```
ticket2 = Ticket(2, "prueba", "media") print(ticket2.mostrar_id())  
print(ticket2.mostrar_tipo()) print(ticket2.mostrar_prioridad())  
print(ticket2.mostrar_estado())
```

#3. crear una clase empleado con atributo nombre

```
#Clase padre class Empleado: def init(self, nombre): self.nombre = nombre def  
trabajar_en_ticket(self, ticket): print(f"El empleado {self.nombre} está trabajando en el  
ticket {ticket.id}.")
```

```
class desarrollador(Empleado): def trabajar_en_ticket(self, ticket): if ticket.tipo ==  
"software": ticket.estado = "resuelto" print (f"el ticket {ticket.id} ha sido asignado al  
desarrollador {self.nombre}.")
```

```
class tester(Empleado): def trabajar_en_ticket(self, ticket): if ticket.tipo == "prueba":  
ticket.estado = "resuelto" print (f"el ticket {ticket.id} ha sido asignado al tester  
{self.nombre}.")
```

```
class projectmanager(Empleado): def asignar_ticket(self, ticket, empleado):  
print(f"{self.nombre} ha asignado el ticket {ticket.id} al empleado {empleado.nombre}.")  
empleado.trabajar_en_ticket(ticket)
```

#crear tickets y empleado (Instancias de objetos)

```
ticket1 = Ticket(1, "software", "alta") ticket2 = Ticket(2, "prueba", "media") ticket3 =  
Ticket(3, "software", "baja") ticket4 = Ticket(4, "prueba", "baja")
```

```
developer1 = desarrollador("Carlitos") tester1 = tester("Juanillo") projectmanager1 =  
projectmanager("Marianita")
```

```
projectmanager1.asignar_ticket(ticket1, developer1)  
projectmanager1.asignar_ticket(ticket2, tester1)
```