

la clase de centro en aprender a usar las funciones y llamarlas para poder optimizar el código use de base los ejercicios que nos dio de tarea para para modificarlos y crear funciones que posteriormente llame

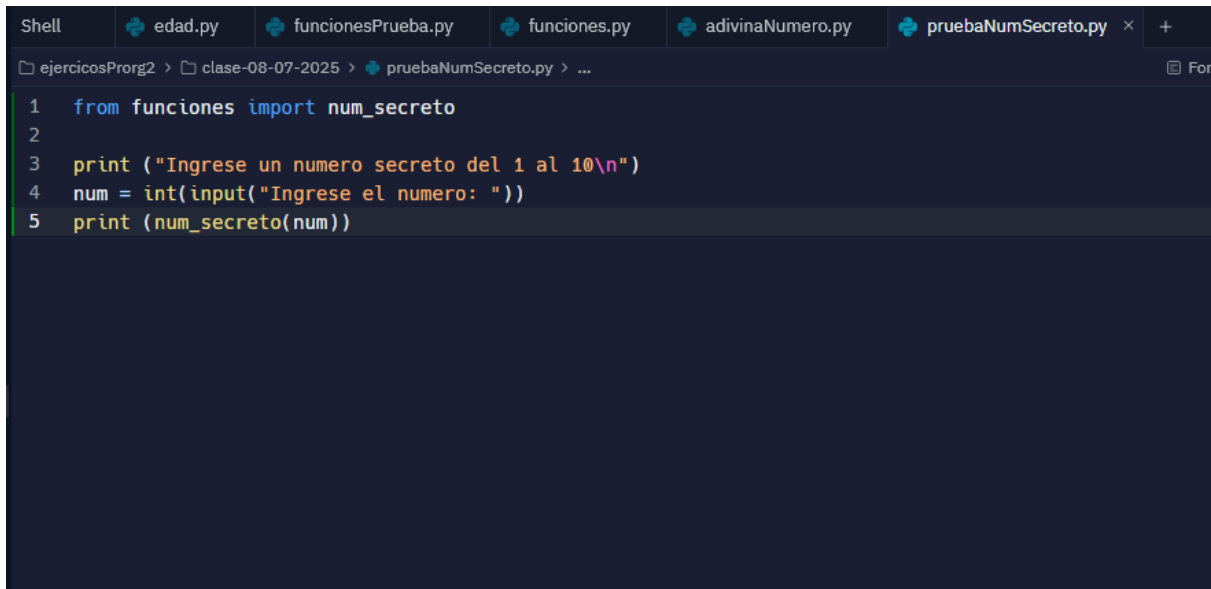
## Versión original de la tarea

```
Shell  edad.py  funcionesPrueba.py  funciones.py  adivinaNumero.py x  pruebaNumSecreto.py  +  ⋮
ejerciciosProrg2 > acti1 > adivinaNumero.py > ...  Format
1  #Adivina el numero secreto
2
3  num = 7
4
5  print("Adivina el numero secreto del 1 al 10\n")
6  num_user = int(input("\nIngrese el numero: "))
7  if num_user == num:
8      print("\nFelicitades adivinaste el numero secreto")
9  else:
10     print("\nLo siento no adivinaste el numero secreto\n")
11
12  print ("____Alejandro Hurtado____")
13
14
```

## Función creada a partir el código original

```
17
18  #Adivina el numero secreto
19
20  def num_secreto(num):
21      print("Adivina el numero secreto del 1 al 10\n")
22      num_user = int(input("Ingrese el numero: "))
23      if num_user == num:
24          return "\nFelicitades, adivinaste el numero secreto"
25      else:
26          return "\nLo siento, no adivinaste el numero secreto\n"
27
28  print ("____Alejandro Hurtado____")
```

## Prueba llamando a la función desde otro archivo de código de Python



The screenshot shows a code editor with several tabs: Shell, edad.py, funcionesPrueba.py, funciones.py, adivinaNumero.py, and pruebaNumSecreto.py. The active tab is pruebaNumSecreto.py, which contains the following Python code:

```
1 from funciones import num_secreto
2
3 print ("Ingrese un numero secreto del 1 al 10\n")
4 num = int(input("Ingrese el numero: "))
5 print (num_secreto(num))
```

## Demostración del Funcionamiento en la terminal de Replit



The screenshot shows a terminal window with the following commands and output:

```
~/workspace/ejercicosProrg2/clase-08-07-2025: python pruebaNumSecreto.py
~/.../ejercicosProrg2/clase-08-07-2025$ python pruebaNumSecreto.py
____Alejandro Hurtado____
Ingrese un numero secreto del 1 al 10

Ingrese el numero: 3
Adivina el numero secreto del 1 al 10

Ingrese el numero: 7

Lo siento, no adivinaste el numero secreto
~/.../ejercicosProrg2/clase-08-07-2025$
```

Realice exactamente el mismo procedimiento con mi otro código que hicimos en clases de comprobar la edad de un cliente para ver qué categoría de películas puede ver

### Código Original:

```
Shell | edad.py | funcionesPrueba.py | funciones.py | adivinaNumero.py | pruebaNumSecreto.py | +
ejerciciosProrg2 > clase-08-07-2025 > edad.py > ...

1  # Alejandro Hurtado
2  def clas_movie(var1):
3      if var1 < 0:
4          return "Edad no valida"
5      elif var1 < 13:
6          return "Te recomendamos peliculas clasificaion G o PG"
7      elif var1 < 18:
8          return "Podes ver peliculas clasificaion PG-13"
9      else:
10         return "Puedes ver cualquier pelicula (Clisificaion R)"
11
12
13 # Pruebas Unitarias para el programa edad.py
14 assert clas_movie(12) == "Te recomendamos peliculas clasificaion G o PG", "Error de edad 12"
15 assert clas_movie(15) == "Podes ver peliculas clasificaion PG-13", "Error de edad 15"
16 assert clas_movie(20) == "Puedes ver cualquier pelicula (Clisificaion R)", "Error de edad 20"
17 assert clas_movie(-5) == "Edad no valida", "Error de edad Negativa"
18
19 # Fin de la funcion clas_movie

20 while True:
21     try:
22         edad = int(input("Ingrese su edad: "))
23         resultado = clas_movie(edad)
24         print(resultado)
25     except ValueError:
26         print("Error: Ingrese un numero entero valido")
27         continue
28
29     continuar = input("¿verificar otra edad? (s/n): ").strip().lower()
30     if continuar != "s":
31         break
32
33 print("Fin del programa")
34 print("____Alejandro Hurtado____")
35
36
```

## Funcion basada en el código

```
Shell | edad.py | funcionesPrueba.py | funciones.py x | adivinaNumero.py | pruebaNumSecreto.py | +
ejercicosProrg2 > clase-08-07-2025 > funciones.py > ...
1 def clas_movie(var1):
2     if var1 < 0:
3         return "Edad no valida"
4     elif var1 < 13:
5         return "Te recomendamos peliculas clasificaion G o PG"
6     elif var1 < 18:
7         return "Podes ver peliculas clasificaion PG-13"
8     else:
9         return "Puedes ver cualquier pelicula (Clisificaion R)"
10
11
12 # Pruebas Unitarias para el programa edad.py
13 assert clas_movie(12) == "Te recomendamos peliculas clasificaion G o PG", "Error de edad 12"
14 assert clas_movie(15) == "Podes ver peliculas clasificaion PG-13", "Error de edad 15"
15 assert clas_movie(20) == "Puedes ver cualquier pelicula (Clisificaion R)", "Error de edad 20"
16 assert clas_movie(-5) == "Edad no valida", "Error de edad Negativa"
17
```

## Llamando a la función

```
Shell | edad.py | funcionesPrueba.py x | funciones.py | adivinaNumero.py | pruebaNumSecreto.py | +
ejercicosProrg2 > clase-08-07-2025 > funcionesPrueba.py > ...
1 from funciones import clas_movie
2
3 while True:
4     try:
5         edad = int(input("Ingrese su edad: "))
6         resultado = clas_movie(edad)
7         print(resultado)
8     except ValueError:
9         print("Error: Ingrese un numero entero valido")
10        continue
11
12    continuar = input("¿verificar otra edad? (s/n): ").strip().lower()
13    if continuar != "s":
14        break
15
16 print("Fin del programa")
17 print("____Alejandro Hurtado____")
```