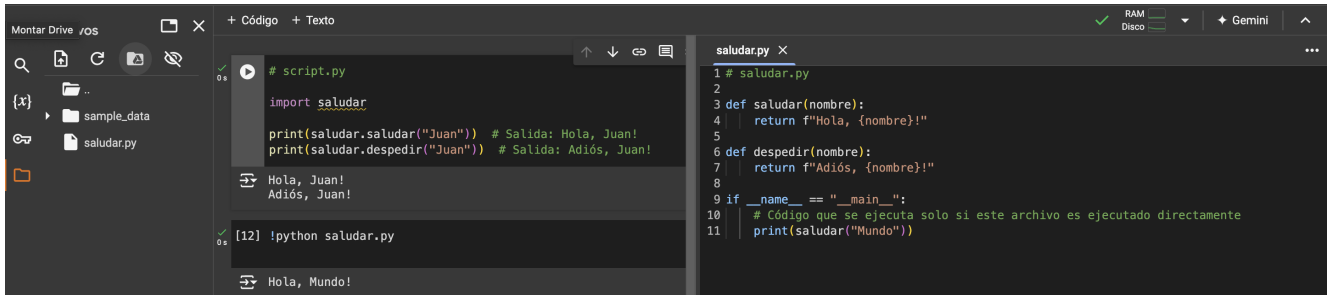


Módulos en Python



Estructura de un módulo en Python

Un módulo en Python es simplemente un archivo con extensión `.py` que contiene definiciones de funciones, clases y variables, así como el código que se ejecuta al importar el módulo. Por ejemplo, supongamos que tienes un archivo llamado `mimodulo.py` con la siguiente estructura:

```
# saludar.py

def saludar(nombre):
    return f"Hola, {nombre}!"

def despedir(nombre):
    return f"Adiós, {nombre}!"

if __name__ == "__main__":
    # Código que se ejecuta solo si este archivo es ejecutado
    directamente
    print(saludar("Mundo"))
```

Importar y usar funciones de un módulo

Cuando importas un módulo en Python, puedes acceder a las funciones y variables definidas en él. Por ejemplo:

```
# script.py

import saludar

print(mimodulo.saludar("Juan")) # Salida: Hola, Juan!
print(mimodulo.despedir("Juan")) # Salida: Adiós, Juan!
```

Cómo funciona el proceso de importación

Cuando Python importa un módulo, sigue estos pasos:

- **Localización del módulo:** Python busca el archivo del módulo en los directorios especificados en la variable `sys.path`.
- **Carga del módulo:** Si el módulo no está ya cargado, Python lo lee y ejecuta.
- **Ejecutar el código del módulo:** Python ejecuta el código en el módulo, lo que incluye la definición de funciones y clases.
- **Importar funciones y variables:** Una vez cargado, puedes acceder a las funciones y variables del módulo usando la notación `mimodulo.funcion()` o `mimodulo.variable`.