

**Índice**

Diseño de pruebas



**Introducción**

* Diseñar pruebas para cada módulo.
* Escribir código que permite probar los métodos.



# Diseño de pruebas



## Pruebas por módulo

### En nuestra solución:

* El programa principal es main.py .
* main.py importa otros módulos.
* Queremos escribir código que nos permita verificar si nuestras definiciones y métodos funcionan correctamente.
* Podemos agregar código al final de cada módulo.



Archivo platos.py

class Plato:

...

class Bebestible(Plato):

...

class Comestible(Plato):

...

## Módulos

Archivo personas.py

class Persona:

...

class Cliente(Persona):

...

class Cocinero(Persona):

...

... # Toma decisiones en base a tipos de plato

Archivo restaurante.py

class Restaurante:

...



**Clase Plato**

class Plato:

...

class Bebestible(Plato):

...

class Comestible(Plato):

...

if name == " main ": plato1 = Bebestible("Jugo")

plato2 = Comestible("Lomo a lo pobre")

plato3 = Bebestible("Pisco Sour") plato4 = Comestible("Empanadas") bebestibles = [plato1, plato3] comestibles = [plato2, plato4] for plato in bebestibles:

print(f"Nombre: {plato.nombre}, Tamaño:

{plato.tamanho}, dificultad: {plato.dificultad}") for plato in comestibles:

print(f"Nombre: {plato.nombre}, Tamaño:

{plato.tiene\_cubiertos}, dificultad: {plato.dificultad}")

* Si agregamos código de prueba al final de la definición de las clases, éste código se ejecutará cada vez que importemos platos.py
* Si usamos la variable name podemos restringir cuando se ejecuta este código de prueba.



## Archivo personas.py

import ... class Persona:

...

class Cliente(Persona):

...

class Cocinero(Persona):

...

if name == " main ":

cliente1 = Cliente("Raul", 7, 30, ["Pisco Sour", "Barros Luco"]) cliente2 = Cliente("Marcela", 9, 10, ["Jugo", "Lomo a lo pobre"]) cocinero = Cocinero("Carla", 15)

print(f"Cliente: {cliente1.nombre}, Exigencia:

{cliente1.exigencia}, Distancia: {cliente1.distancia}") print(f"Cocinero: {cocinero.nombre}, Energia:

{cocinero.energia}")

plato\_resultado1 = cocinero.cocinar(["Limonada", "Bebestible"]) plato\_resultado2 = cocinero.cocinar(["Barros Luco",

"Comestible"])

print(f"Nombre plato: {plato\_resultado1.nombre}, Calidad:

{plato\_resultado1.calidad}")

print(f"Nombre plato: {plato\_resultado2.nombre}, Calidad:

{plato\_resultado2.calidad}")

print(f"Cocinero: {cocinero.nombre}, Energia:

{cocinero.energia}")

### Podemos:

* Crear algunos clientes con los parámetros que necesitan de acuerdo a su método inicializador.
* Crear un cocinero, hacer que cocine algunos platos, e imprimir su estado antes y después de prepararlos, para verificar que su energía ha disminuido.
* Hacer que imprima también el resultado de calidad de los platos.



## Síntesis

* Agregar código al final de cada módulo.
* El código agregado nos permite hacer pruebas individuales de cada módulo.

## Referencias bibliográficas

* Documentación Python 3.8. https://docs.python.org/3/reference