OBJECT ORIENTED PROGRAMMING WITH PYTHON

Second Class

1st Semester

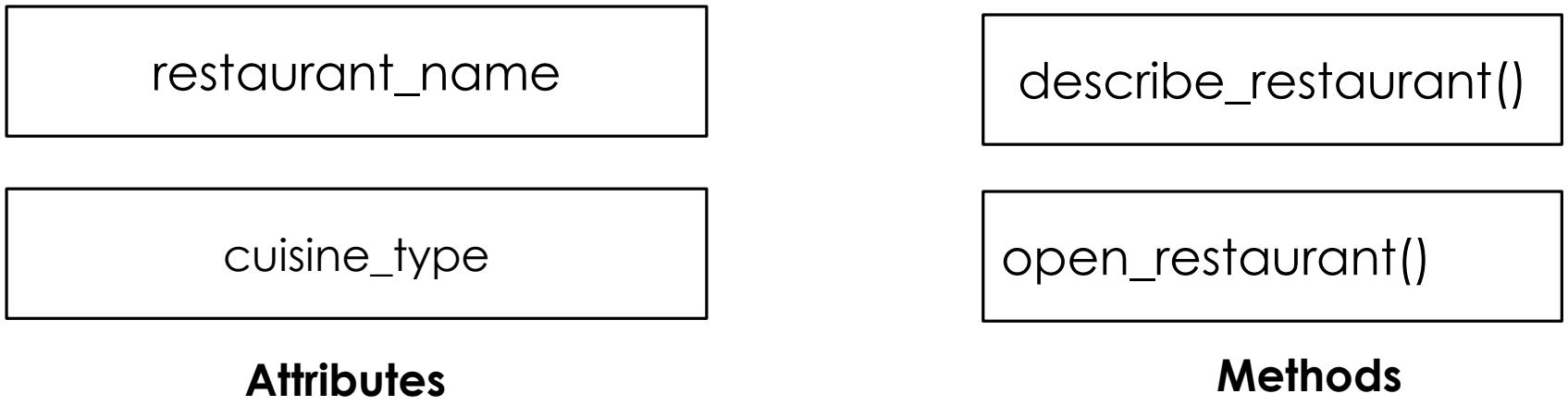
EXAMPLES ON CLASSES AND OBJECTS

- **Restaurant**: Make a class called Restaurant.
- The **__init__()** method for Restaurant should store **two attributes**:
  a **restaurant_name** and a **cuisine_type**.
- Make a **method** called
  describe_restaurant() that prints these two pieces of information,
  open_restaurant() that prints a message indicating that the restaurant is open.
- Create three different instances from the class,(**asian_rest, euro_rest, kfc_rest**).
- Call **describe_restaurant()** method for each instance.

CREATE RESTAURANT CLASS

Example 1 : Create Restaurant Class
class : Restaurant
attributes : restaurant_name, and cuisine_type
methods : describe_restaurant() and open_restaurant()

objects : **asian_rest, euro_rest, kfc_rest**



RESTAURANT EXAMPLE

```
class Restaurant:
    def __init__(self, restaurant_name, cuisine_type):
        self.restaurant_name = restaurant_name
        self.cuisine_type = cuisine_type

    def discribe_restaurant(self):
        print(f""Welcome to {self.restaurant_name} Restaurant and the delicious dishes frome the
{self.cuisine_type} Cuisine .."")

    def open_restaurant(self):
        print(" our Restaurant is open Now ^_^ ")


asian_rest = Restaurant('Shanghai' , 'Asian')
asian_rest.discribe_restaurant()
asian_rest.open_restaurant()
euro_rest =Restaurant ('Big Pizza' , 'Italian')
euro_rest.discribe_restaurant()
euro_rest.open_restaurant()
kfc_rest=Restaurant('KFC' ,'Fast Food')
kfc_rest.discribe_restaurant()
```

IMPROVED RESTAURANT EXAMPLE

Now Start with your class from the previous Exercise

- Add an attribute called number served with a default value of 0.
- Create an instance called **kfc_rest** from this class.
- Print the number of customers the restaurant has served,
- Add a method called **set_number_served()** that lets you set the number of customers that have been served.
- Call this method with a new number and print the value again
- Add a method called **increment_number_served()** that lets you incrementthe number of customers who've been served.
- Call this method with any number you like that could represent how many customers were served in.

CREATE RESTAURANT CLASS

```
class Restaurant:
    def __init__(self, restaurant_name, cuisine_type):
        self.restaurant_name = restaurant_name
        self.cuisine_type = cuisine_type
        self.number = 0

    def set_number_served(self, number):
        self.number = number

    def increment_number_served(self, add):
        self.number += add

    def discribe_restaurant(self):
        print(f""Welcome to {self.restaurant_name} Restaurant and the delicious dishes frome the {self.cuisine_type} Cuisine .."")

    def open_restaurant(self):
        print(" our Restaurant is open Now ^_^ ")


kfc_rest=Restaurant('KFC' ,'Fast Food')
print(f'The number of serverd customres in {kfc_rest.restaurant_name} are {kfc_rest.number}')
kfc_rest.discribe_restaurant()
kfc_rest.set_number_served(15)
print(f'The number of serverd customres in {kfc_rest.restaurant_name} are {kfc_rest.number} ')
kfc_rest.increment_number_served(5)
print(f'The number of serverd customres in {kfc_rest.restaurant_name} are {kfc_rest.number} ')
```

ICE CREAM STAND EXAMPLE

An ice cream stand is a specific kind of restaurant.

- Write a class called **IceCreamStand** that inherits from the Restaurant class you wrote In the Restaurant Exercise
- Add an attribute called **flavors** that stores a list of ice cream flavors.
- Write a method call it **display_flav()** that displays these flavors.
- Create an instance of **IceCreamStand**, and call this method.

CREATE RESTAURANT CLASS

```
class Restaurant:
    def __init__(self, restaurant_name, cuisine_type):
        self.restaurant_name = restaurant_name
        self.cuisine_type = cuisine_type
        self.number = 0

    def set_number_served(self, number):
        self.number = number

    def increment_number_served(self, add):
        self.number += add

    def discribe_restaurant(self):
        print(f"""Welcome to {self.restaurant_name} Restaurant and the delicious dishes frome the {self.cuisine_type} Cuisine .."")

    def open_restaurant(self):
        print(" our Restaurant is open Now ^_^ ")

class IceCreamStand(Restaurant):
    def __init__(self, restaurant_name,cuisine_type):
        super().__init__(restaurant_name,cuisine_type)
        self.flavors=['strawberry','peach','apple','watermelon']
        self.stores='MADO'

    def display_flav(self):
        print(f" hi our {self.stores} store has four flavours:")
        print(self.flavors)

icecreram=IceCreamStand('HappyDay','IceCream')
icecreram.discribe_restaurant()
icecreram.display_flav()
```

Very important to add more attributes and methods