Day – 6 Morning Assessment

Instance Method Questions

1. Instance Method is a method where we create an object for the class to access the class attributes and methods.

Eg:

class Student:

def \_\_init\_\_(self,name,gender):

self.name = name

self.gender = gender

def display(self):

print(f”Student’s name is {self.name}”)

print(f”Student;s gender is {self.gender}”)

stu = Student(“Purvi”, “Female”)

stu.display()

1. class Car:

def \_\_init\_\_(self,name):

self.name = name

def start\_engine(self):

print(f”{self.name} engine started”)

car = Car(“Bugatti”)

car.start\_engine()

1. We call the instance method with the object.

Eg:

class Car:

def \_\_init\_\_(self,name):

self.name = name

def start\_engine(self):

print(f”{self.name} engine started”)

car = Car(“Bugatti”) #created object for the class

car.start\_engine() #calling the method with the object created

1. class Circle:

def \_\_init\_\_(self, radius):

self.radius = radius

def area(self):

print(f” The area of the given circle is {2\*3.14\*radius}”)

cir = Circle(5)

cir.area()

1. Error arises as there will be no self attribute passed over to \_\_init\_\_. We need to create an object to pass the self attribute to the \_\_init\_\_

Class Method Questions

1. Class method is a method that are directly associated with the class rather than the instance of the class.

Eg:

class Student:

school\_name = “ABC School”

def \_\_init\_\_(self,name):

self.name = name

@classmethod

def change\_name(cls, new\_name):

cls.school\_name = new\_name

stu = Student(“Priya”)

Student.change\_name(“BCD School”)

stu.change\_name()

1. Class Employee:

company\_name = “TechCorp”

def \_\_init\_\_(self,name):

self.name = name

@classmethod

def change\_name(cls, new\_name):

cls.company\_name = new\_name

emp = Employee(“Priya”)

Employee.change\_name(“BCD School”)

emp.change\_name()

1. We define a class method using the annotation called “@classmethod”
2. Class Counter:

def \_\_init\_\_(self,count):

self.count = count

def increment(self):

self.count +=1

print(f”The count is {self.count})

cou = Counter(0)

cou.increment()

Static Method:

1. Static method is where the method is independent of any instance of the class, so it can be accessed using the class name.

Eg:

class Sum:

@staticmethod

def static\_method(x,y):

return f”sum of {x} and {y} is {x+y}”

print(Sum.static\_method(4,10))

s = Sum()

print(s.static\_method(7,9)

1. class MathUtils:

@staticmethod

def is\_even(num):

if num%2 == 0:

return True

else:

return False

print(MathUtils.is\_even(45))

1. We have to use Class method type if we want to access/modify class variables.

We have to use Instance method type if we want to access/modify instance variables.