[Solution] CP04

July 18, 2022

1 Class Partipation 04

1.0.1 01. Employee Class

Write a class named Employee that holds the following data about an employee in attributes: name, ID number, department, and job title.

Once you have written the class, write a program that creates three Employee objects to hold the following data:

Name	ID	Number	Department	Job Title
Susan Meyers	47899		Accounting	Vice President
Mark Jones	39119		IT	Programmer
Joy Rogers	81774		Manufacturing	Engineer

he program should store this data in the three objects and then display the data for each employee on the screen.

[1]:
```
class Employee:

    def __init__(self, name="Addison", id_num="123", dept="IT",␣
↪job_title="Manager"):
        self.__name = name
        self.__id_num = id_num
        self.__dept = dept
        self.__job_title = job_title

    def set_name(self, name):
        self.__name = name

    def get_name(self):
        return self.__name

    def set_id_num(self, id_num):
        self.__id_num = id_num

    def get_id_num(self):
        return self.__id_num

    def set_dept(self, dept):
        self.__dept = dept

    def get_dept(self):
        return self.__dept

    def set_job_title(self, job_title):
        self.__job_title = job_title

    def get_job_title(self):
        return self.__job_title

    def __str__(self):
        return "Name: {}, ID Number: {}, Department: {}, Job Title: {}".
 ↪format(self.get_name(), self.get_id_num(),
                                                                              ␣
 ↪self.get_dept(), self.get_job_title())


e1 = Employee('Susan Meyers', 47899, 'Accounting', 'Vice President')
e2 = Employee('Mark Jones', 39119, 'IT', 'Programmer')
e3 = Employee('Joy Rogers', 81774, 'Manufacturing', 'Engineer')

print(e1)
print(e2)
print(e3)
```

```
Name: Susan Meyers, ID Number: 47899, Department: Accounting, Job Title: Vice
President
Name: Mark Jones, ID Number: 39119, Department: IT, Job Title: Programmer
Name: Joy Rogers, ID Number: 81774, Department: Manufacturing, Job Title:
Engineer
```

1.0.2 02. Employee and ProductionWorker Classes

Write an Employee class that keeps data attributes for the following pieces of information: * Employee name * Employee number

Next, write a class named ProductionWorker that is a subclass of the Employee class. The ProductionWorker class should keep data attributes for the following information:

- Shift number (an integer, such as 1, 2, or 3)
- Hourly pay rate

The workday is divided into two shifts: day and night. The shift attribute will hold an integer value representing the shift that the employee works. The day shift is shift 1 and the night shift is shift 2. Write the appropriate accessor and mutator methods for each class.

Once you have written the classes, write a program that creates an object of the ProductionWorker class and prompts the user to enter data for each of the object's data attributes. Store the data in the object and then use the object's accessor methods to retrieve it and display it on the screen.

[2]:

```python
class Employee:

    def __init__(self, name="Addison", number="123"):
        self.__name = name
        self.__number = number

    def set_name(self, name):
        self.__name = name

    def get_name(self):
        return self.__name

    def set_number(self, number):
        self.__number = number

    def get_number(self):
        return self.__number

    def __str__(self):
        return "Name: {}, Number: {}".format(self.get_name(), str(self.
 ↪get_number()))
```

[3]:

```python
class ProductionWorker(Employee):
    def __init__(self, name="Addison", number="123", shift_number=0,␣
 ↪hourly_rate=100):
        Employee.__init__(self, name, number)
        self.shift_number = 0
        self.__hourlyPayRate = 100

    def set_shift_number(self, shift_number):
        self.__shift_number = shift_number

    def get_shift_number(self):
        if self.__shift_number == 1:
            return "Day"
        elif self.__shift_number == 2:
            return "Night"
        else:
            return "wrong Entry"

    def set_hourly_rate(self, hourly_rate):
        self.__hourly_rate = hourly_rate
    def get_hourly_rate(self):
        return self.__hourly_rate
```

[4]:

```
obj = ProductionWorker()

obj.set_name(input("Enter name : "))
obj.set_number(input("Enter number : "))
obj.set_shift_number(int(input("Enter Shift (1/2) : ")))
obj.set_hourly_rate(input("Enter hourly rate : "))

print("Name : ", obj.get_name())
print("Number : ", obj.get_number())
print("Shift : ", obj.get_shift_number())
print("Hourly Rate : ", obj.get_hourly_rate())
```

```
Enter name : Addison
Enter number : 159
Enter Shift (1/2) : 2
Enter hourly rate : 158
Name :  Addison
Number :  159
Shift :  Night
Hourly Rate :  158
```

1.0.3 03. Person and Customer Classes

Write a class named Person with data attributes for a person's name, address, and telephone number. Next, write a class named Customer that is a subclass of the Person class. The Customer class should have a data attribute for a customer number and a Boolean data attribute indicating whether the customer wishes to be on a mailing list. Demonstrate an instance of the Customer class in a simple program.

[5]:

```
class Person:
    def __init__(self):
        self.__name = ""
        self.__address = ""
        self.__telephone = ""

    def set_name(self, name):
        self.__name = name

    def set_address(self, add):
        self.__address = add

    def set_tel(self, tel):
        self.__telephone = tel

    def get_name(self):
```

```python
        return self.__name

    def get_address(self):
        return self.__address

    def get_tel(self):
        return self.__telephone
```

[6]:

```python
class Customer(Person):
    def __init__(self):
        Person.__init__(self)
        self.__customer_number = ""
        self.__mailing_list = -1

    def set_number(self, number):
        self.__customer_number = number

    def set_mailing(self, mailing):
        self.__mailing_list = mailing

    def get_number(self):
        return self.__customer_number

    def get_mailing(self):
        if self.__mailing_list == 0:
            return False
        else:
            return True
```

[7]:

```python
customer = Customer()
customer.set_name("John")
customer.set_tel("03331")
customer.set_number("1203")
customer.set_address("NY")
customer.set_mailing(1)

print("Name : ", customer.get_name())
print("Telephone : ", customer.get_tel())
print("Number : ", customer.get_number())
print("Address : ", customer.get_address())
print("Want to be in mailing list : ", customer.get_mailing())
```

```
Name :  John
Telephone :  03331
Number :  1203
Address :  NY
Want to be in mailing list :  True
```

[8]:

```python
class Customer(Person):
    def __init__(self):
        Person.__init__(self)
        self.__customerNumber=""
        self.__mailingList=-1
    def set_number(self,number):
        self.__customerNumber = number
    def set_mailing(self , mailing):
        self.__mailingList = mailing
    def get_number(self):
        return self.__customerNumber
    def get_mailing(self):
        if self.__mailingList == 0:
            return False
        else:
            return True
```

[9]:

```python
customer = Customer()
customer.set_name("John")
customer.set_tel("03331")
customer.set_number("1203")
customer.set_address("NY")
customer.set_mailing(1)

print("Name : ",customer.get_name())
print("Telephone : ",customer.get_tel())
print("Number : ",customer.get_number())
print("Address : ",customer.get_address())
print("Want to be in mailing list : ",customer.get_mailing())
```

```
Name :  John
Telephone :  03331
Number :  1203
Address :  NY
Want to be in mailing list :  True
```