LAB 7

TASK 1

|  |
| --- |
| Fan |
| -speed: int  -radius: float  -color: string  -on: bool |
| Fan(speed=SLOW: int, radius=5.0: float, color=blue: string, on=False: bool): None  get\_speed(): int  set\_speed(speed: int): None  get\_radius(): float  set\_radius(radius: float): None  get\_color(): string  set\_color(color: string): None  get\_on(): bool  set\_on(on: bool): None |

TASK 2

class Fan:

SLOW = 1

MEDIUM = 2

FAST = 3

def \_\_init\_\_(self, speed=SLOW, radius=5.0, color="blue", on=False):

self.\_\_speed = speed

self.\_\_radius = radius

self.\_\_on = on

self.\_\_color = color

def get\_speed(self):

return self.\_\_speed

def get\_radius(self):

return self.\_\_radius

def get\_on(self):

return self.\_\_on

def get\_color(self):

return self.\_\_color

def set\_speed(self, new\_speed):

if isinstance(new\_speed, int) and new\_speed >= 1 and new\_speed <= 3:

self.\_\_speed = new\_speed

else:

print("enter valid data")

def set\_color(self, new\_color):

if isinstance(new\_color, str):

self.\_\_color = new\_color

else:

print("enter valid data")

def set\_radius(self, new\_radius):

if isinstance(new\_radius, float):

self.\_\_radius = new\_radius

else:

print("enter valid data")

def set\_on(self, new\_on):

if isinstance(new\_on, bool):

self.\_\_on = new\_on

else:

print("enter valid data")

def main():

fan1 = Fan()

fan1.set\_speed(Fan.FAST)

fan1.set\_radius(10.0)

fan1.set\_color("yellow")

fan1.set\_on(True)

fan2 = Fan()

fan2.set\_speed(Fan.MEDIUM)

fan2.set\_radius(5.0)

fan2.set\_color("blue")

fan2.set\_on(False)

print(fan1.get\_speed(), fan1.get\_radius(), fan1.get\_color(), fan1.get\_on())

print(fan2.get\_speed(), fan2.get\_radius(), fan2.get\_color(), fan2.get\_on())

main()

TASK 3

|  |
| --- |
| Account |
| -idd: int  -balance: float  -annual\_interest\_rate: float |
| Account(idd=0: int, balance=100: float, annual\_interest\_rate=0: float): None  get\_idd(): int  get\_balance(): float  get\_annual\_interest\_rate(): float  set\_idd(idd: int): None  set\_balance(balance: float): None  set\_annual\_interest\_rate(annual\_interest\_rate: float): None  get\_monthly\_interest\_rate(): float  get\_monthly\_interest(): float  with\_draw(amount: int): None  deposit(amount: int): None |

TASK 4

class Account:

def \_\_init\_\_(self, idd=0, balance = 100, annual\_interest\_rate=0):

self.\_\_idd = idd

self.\_\_balance = balance

self.\_\_annual\_interest\_rate = annual\_interest\_rate

def get\_idd(self):

return self.\_\_idd

def get\_balance(self):

return self.\_\_balance

def get\_annual\_interest\_rate(self):

return self.\_\_annual\_interest\_rate

def set\_idd(self, new\_idd):

if isinstance(new\_idd, int):

self.\_\_idd = new\_idd

else:

print("Enter valid data")

def set\_balance(self, new\_balance):

if isinstance(new\_balance, float):

self.\_\_balance = new\_balance

else:

print("Enter valid data")

def set\_annual\_interest\_rate(self, new\_annual\_interest\_rate):

if isinstance(new\_annual\_interest\_rate, float):

self.\_\_annual\_interest\_rate = new\_annual\_interest\_rate

else:

print("Enter valid data")

def get\_monthly\_interest\_rate(self):

monthly\_interest\_rate = (self.\_\_annual\_interest\_rate/12)/100

return monthly\_interest\_rate

def get\_monthly\_interest(self):

monthly\_interest = self.\_\_balance\*self.get\_monthly\_interest\_rate()

return monthly\_interest

def with\_draw(self, amount):

self.\_\_balance - amount

def deposit(self, amount):

self.\_\_balance + amount

def main():

account1 = Account()

account1.set\_idd(1122)

account1.set\_balance(20000.0)

account1.set\_annual\_interest\_rate(4.5)

account1.with\_draw(2500)

account1.deposit(3000)

print(account1.get\_idd(), ",", account1.get\_balance(), ",", account1.get\_monthly\_interest\_rate(), ",", account1.get\_monthly\_interest())

main()