

Національний технічний університет України «Київський
політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Програмування

Лабораторна робота №6

«Основи об'єктно-орієнтованого програмування. Модулі та
пакети»

Виконав:
студент групи ІО-41
Давидчук А. М.
Залікова книжка № 4108

Перевірив
Пономаренко А. М.

Тема: «Основи об'єктно-орієнтованого програмування. Модулі та пакети».

Мета: вивчити способи створення та підключення модулів та пакетів. Основи ООП. Методи та атрибути класів та робота з ними. Побудова програми у стилі ООП.

Короткі теоретичні відомості:

class – оператор створення класу;
def __init__ – конструктор класу – метод, який автоматично викликається при створенні екземплярів;
self – об'єкт екземпляра;
density – метод класу;
round – вбудована функція округлення;
while – цикл “поки”;
planet.name – атрибути екземпляра;
input – вбудована функція вводу значень;
for – оператор циклу, кількість проходження якого залежить від кількості елементів певного ітерованого об'єкта;
if – оператор умовного розгалуження;
print() – вбудована функція виводу.
sorted() – вбудована функція сортування.

Завдання:

| | |
|---|--|
| 8 | Створити клас, який описує планету Сонячної системи. Цей клас повинен містити назву планети, її порядковий номер, рахуючи від Сонця, відстань від Сонця, об'єм та масу, орбітальну швидкість. Методи класу мають визначати густину планети, відношення маси Землі до маси даної планети, відношення орбітальної швидкості Землі до о.ш. планети. Створити об'єкти даного класу для планет Сонячної системи та ввести дані, користуючись даними з Вікіпедії. Програма повинна обчислювати та виводити за запитом густину планети, відношення маси та о.ш. Землі до маси та о.ш. планети, введеної за запитом. Програма повинна також виводити за запитом список планет у порядку зростання їх густини з зазначенням номеру планети та її орбітальної швидкості. |
|---|--|

Код:

```
class Planets:

    def __init__(self, name="", number=0, distance=0, volume=0, mass=0, speed=0):

        self.name = name
        self.number = number
        self.distance = distance
        self.volume = volume
        self.mass = mass
        self.speed = speed

    def density(self):
```

```

        return round(self.mass / self.volume, 4)

    def mass_ratio(self):

        return round(5.976 / self.mass, 4)

    def speed_ratio(self):

        return round(29.78 / self.speed, 4)

mercury = Planets(name="Меркурій", number=1, distance=57.9, volume=3.7,
mass=3.303, speed=15.06)
venus = Planets(name="Венера", number=2, distance=108.2, volume=12.11,
mass=4.867, speed=24.62)
earth = Planets(name="Земля", number=3, distance=149.6, volume=12.12,
mass=5.976, speed=29.78)
mars = Planets(name="Марс", number=4, distance=227.9, volume=6.419,
mass=6.417, speed=24.62)
jupiter = Planets(name="Юпітер", number=5, distance=778.6, volume=13.08,
mass=1898.0, speed=11.86)
saturn = Planets(name="Сатурн", number=6, distance=1433.5, volume=9.44,
mass=568.0, speed=29.78)
uranus = Planets(name="Уран", number=7, distance=2872.5, volume=8.69,
mass=86.8, speed=82.74)
neptune = Planets(name="Нептун", number=8, distance=4498.1, volume=11.15,
mass=102.4, speed=164.79)
pluto = Planets(name="Плутон", number=9, distance=5906.4, volume=0.91,
mass=206.2, speed=24.62)

planet_list = [mercury, venus, earth, mars, jupiter, saturn, uranus,
neptune, pluto]
result = []

planet_name = input("Введіть планету для розрахунку: ")
for planet in planet_list:

    if planet.name == planet_name:

        print(f"Інформація щодо {planet.name}:\n",
              f"Густина = {planet.density()} кг/м3\n",
              f"Маса {planet.name} відносно маси Землі =
{planet.mass_ratio()}\n",
              f"Швидкість {planet.name} відносно швидкості Землі =
{planet.speed_ratio()}\n")

for x in sorted([planet.density() for planet in planet_list]):

    for planet in planet_list:

        if planet.density() == x: result.append(planet.name)

print(f"Список планет з найбільш більшою густиною:\n", result)

```

Знімок екрана контрольного прикладу:

```
Введіть планету для розрахунку: Уран
Інформація щодо Уран:
  Густина = 9.9885 кг/м3
  Маса Уран відносно маси Землі = 0.0688
  Швидкість Уран відносно швидкості Землі = 0.3599

Список планет з найбільш більшою густиною:
['Венера', 'Земля', 'Меркурій', 'Марс', 'Нептун', 'Уран', 'Сатурн', 'Юпітер', 'Плутон']

Введіть планету для розрахунку: Земля
Інформація щодо Земля:
  Густина = 0.4931 кг/м3
  Маса Земля відносно маси Землі = 1.0
  Швидкість Земля відносно швидкості Землі = 1.0

Список планет з найбільш більшою густиною:
['Венера', 'Земля', 'Меркурій', 'Марс', 'Нептун', 'Уран', 'Сатурн', 'Юпітер', 'Плутон']
```

Висновок:

Я написав програму з використанням класів. Доволі успішно продемонстрував вірність виконання програм та розуміння концептів ООП, та роботи з ними. Розрахунки відбувались у вкладених циклах та робота з екземплярами. При виводі використав форматування.