

Laboratorium 2

Temat: Python, ćwiczenia rozwojowe.

Kody powstałe w wyniku realizacji zadań laboratoryjnych:

Klasa Student.py

```
class Student:
    licznik = 0

    def __init__(self, imie, nazwisko, numerIndeksu, kierunek):
        Student.licznik += 1
        self.imie = imie
        self.nazwisko = nazwisko
        self.__numerIndeksu = numerIndeksu
        self.kierunek = kierunek

    def __str__(self):
        return "Imię: %s Nazwisko: %s Kierunek: %s Indeks: %s" % (
            self.imie, self.nazwisko, self.kierunek, self.__numerIndeksu)

    def __eq__(self, other):
        if self.imie == other.imie and self.nazwisko == other.nazwisko:
            return True
        else:
            return False

    def __lt__(self, other):
        if self.imie < other.imie and self.nazwisko < other.nazwisko:
            return True
        else:
            return False

    def getLicznik(self):
        return Student.licznik

    def rokDrugi(self):
        print(self.__str__() + ' Rok: 2')

    ## Usprawniona funkcja rok_drugi według mnie powinna przyjmować
    ## dodatkowy parametr
    ## w tym wypadku rok studiów
    ## poprawiona funkcja prezentuje się następująco
    def rok_studiow(self, rokStudiow):
        print(self.__str__() + ' Rok:', rokStudiow)
    # funkcja rok drugi nie jest zła jednak, takie harcodowanie nie jest
    # dobra praktyka
    # starajmy sie aby nasze rozwiązania były reużywalne oraz miały sens

class StudentInformatyki(Student):

    def __init__(self, imie, nazwisko, numerIndeksu, kierunek,
specjalnosc):
        super().__init__(imie, nazwisko, numerIndeksu, kierunek)
        self.kierunek = 'IIS'
        self.specjalnosc = specjalnosc
```

Klasa testowa teststudent.py

```
import Lab2.Student as stud

def testy():
    print('Zadanie 1,Zadanie 3 ,Zadanie 4')
    s1 = stud.Student('Wojciech', 'Pietruszyński', 19443, 'Informatyka')
    # print(student.numer_indeksu) jeśli pole prywatne to wywołanie jest
    # nie możliwe
    # przy obiekcie możemy się odwołać
    print(s1.getLicznik(), s1)
    s2 = stud.StudentInformatyki('Jan', 'Kowalski', 12345, None,
    'Programowanie')
    print(s2.getLicznik(), s2.__str__(), 'Specjalnosc: ', s2.specjalnosc)
    s3 = stud.StudentInformatyki('Jan', 'Nowak', 12346, None,
    'Programowanie')
    print(s3.getLicznik(), s3.__str__(), 'Specjalnosc: ', s3.specjalnosc)
    print('Zadanie 2')
    print(s1.imie, s1.nazwisko, "<", s2.imie, s2.nazwisko)
    print(s1.__lt__(s2))
    print(s1.imie, s1.nazwisko, "<", s3.imie, s3.nazwisko)
    print(s2.__lt__(s1))
    print()
    print(s1.imie, s1.nazwisko, "==", s2.imie, s2.nazwisko)
    print(s1.__eq__(s2))
    print(s1.imie, s1.nazwisko, "==", s3.imie, s3.nazwisko)
    print(s2.__eq__(s1))
    print(s1.imie, " ", s1.nazwisko, "==", s1.imie, s1.nazwisko)
    print(s1.__eq__(s1))

    print("Zadanie 5")
    stud.Student.rokDrugi(s1)

    print("Zadanie 5 poprawiona metoda")
    stud.Student.rok_studiow(s2, 4)

if __name__ == "__main__":
    testy()
```

Wyniki działania powstałych programów :

C:\Users\Wojciech\AppData\Local\Programs\Python\Python38\python.exe

C:/Users/Wojciech/Desktop/Lab_PWJS_19443/Lab2/teststudent.py

Zadanie 1,Zadanie 3 ,Zadanie 4

1 Imię: Wojciech Nazwisko: Pietruszyński Kierunek: Informatyka Indeks: 19443

2 Imię: Jan Nazwisko: Kowalski Kierunek: IIS Indeks: 12345 Specjalnosc: Programowanie

3 Imię: Jan Nazwisko: Nowak Kierunek: IIS Indeks: 12346 Specjalnosc: Programowanie

Zadanie 2

Wojciech Pietruszyński < Jan Kowalski

False

Wojciech Pietruszyński < Jan Nowak

True

Wojciech Pietruszyński == Jan Kowalski

False

Wojciech Pietruszyński == Jan Nowak

False

Wojciech Pietruszyński == Wojciech Pietruszyński

True

Zadanie 5

Imię: Wojciech Nazwisko: Pietruszyński Kierunek: Informatyka Indeks: 19443 Rok: 2

Zadanie 5 poprawiona metoda

Imię: Jan Nazwisko: Kowalski Kierunek: IIS Indeks: 12345 Rok: 4

Process finished with exit code 0