

**Московский государственный технический
университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Базовые компоненты интернет-технологий»

Отчет по рубежному контролю №1

Выполнил:
студент группы
ИУ5И-35Б
Давиташвили Шако

Москва, 2021 г.

Описание задания:

(Вариант предметной области – 23, вариант запросов – Д)

1. «Синтаксическая конструкция» и «Язык программирования» связаны соотношением один-ко-многим. Выведите операторы, которые оканчиваются на 'оп' и сколько дней изучают язык, к которому они относятся
2. «Синтаксическая конструкция» и «Язык программирования» связаны соотношением один-ко-многим. Выведите список языков программирования со средним количеством дней уделенным для изучения каждого из них, отсортированный по среднему количеству дней.
3. «Синтаксическая конструкция» и «Язык программирования» связаны соотношением многие-ко-многим. Выведите список всех языков программирования, у которых имя начинается на «С», и названия их операторов.

Текст программы:

```
from operator import itemgetter
```

```
class SynConstr:
```

```
    """ Синтаксическая конструкция  
    операторы, которые относятся к данному языку  
    количество дней в течение которого изучают этот язык """
```

```
    def __init__(self, id, oper, days, ProgLan_id):
```

```
        self.id = id  
        self.oper = oper  
        self.days = days  
        self.ProgLan_id = ProgLan_id
```

```
class ProgLan:
```

```
    # Язык программирования
```

```
    def __init__(self, id, name):
```

```
        self.id = id  
        self.name = name
```

```
class SCPL:
```

```
    # синтаксическая конструкция языка программирования
```

```
    def __init__(self, ProgLan_id, SynConstr_id):
```

```
        self.ProgLan_id = ProgLan_id  
        self.SynConstr_id = SynConstr_id
```

```
# Языки программирования
```

```
Langs = [
```

```
    ProgLan(1, 'Pyhton'),  
    ProgLan(2, 'Java'),  
    ProgLan(3, 'JavaScript'),
```

```

    ProgLan(4, 'C++'),
    ProgLan(5, 'C#'),
    ProgLan(6, 'R'),
]

```

Синтаксические конструкции и количество дней, в течение которых изучают язык

```

SynConstrs = [
    SynConstr(1, 'Condition', 273, 1),
    SynConstr(2, 'Loop', 381, 2),
    SynConstr(3, 'Shift', 145, 3),
    SynConstr(4, 'Add', 11, 4),
    SynConstr(5, 'Mul', 253, 5),
    SynConstr(6, 'Semicolon', 3, 6),
    SynConstr(7, 'Array', 73, 1),
]

```

#Syntactic Constructions of Programming Languages/синтаксические конструкции языков программирования

```

SCPLs = [
    SCPL(1,2),
    SCPL(2,3),
    SCPL(3,4),
    SCPL(4,5),
    SCPL(5,6),
    SCPL(6,7),
    SCPL(3,2),
    SCPL(4,3),
    SCPL(5,4),
    SCPL(6,5),
    SCPL(5,6),
    SCPL(1,1),
    SCPL(2,2),
    SCPL(3,3),
    SCPL(4,4),
    SCPL(5,5),
    SCPL(6,6),
    SCPL(7,7),
    SCPL(7,3),
    SCPL(2,5),
    SCPL(1,6),
    SCPL(6,4),
]

```

```

def main():
    """Основная функция"""

```

Соединение данных один-ко-многим

```

one_to_many = [(s.oper,s.days,l.name)
    for l in Langs
    for s in SynConstrs
    if l.id==s.ProgLan_id]

```

Соединение данных многие-ко-многим

```

many_to_many = [
    (s.oper, s.days, l.name)
    for l in Langs
    for s in SynConstrs
    for sp in SCPLs
    if l.id == sp.ProgLan_id and l.id == sp.SynConstr_id
]

```

#вывести операторы, которые оканчиваются на 'оп' и сколько дней изучают язык, к которому они относятся

```

print("\nЗаданиеД1")

```

```

res1=list(filter(lambda x : x[0].endswith('оп'), one_to_many))
[print(x[0],x[1]) for x in res1]

```

```

print("\nЗаданиеД2")

```

```

res2_unsorted = []

```

```

for l in Langs:

```

#список языковых конструкций языков программирования

```

l_SynConstrs = list(filter(lambda x: x[2]==l.name, one_to_many))

```

```

if(len(l_SynConstrs)>0):

```

```

    l_days = [days for _,days,_ in l_SynConstrs]

```

```

    l_days_all = sum(l_days)

```

```

    average_days = l_days_all/len(l_days)

```

```

    res2_unsorted.append((l.name, average_days))

```

```

res2 = sorted(res2_unsorted, key=itemgetter(1), reverse=True)
print(res2)

```

```

print("\nЗаданиеД3")

```

```

res3={}

```

```

for l in Langs:

```

```

    if l.name.startswith('C'):

```

#список языковых конструкций языков программирования

```

    l_SynConstrs = list(filter(lambda x: x[2]==l.name, many_to_many))

```

```

    names = [i for i, _, _ in l_SynConstrs]

```

```

    res3[l.name] = names

```

```

print(res3)

```

```

if __name__ == '__main__':

```

```

    main()

```

Результат выполнения программы:

```

[hakodavitahvili@MacBook-Pro-Shako RK1 % python3 rk1.py

```

```

ЗаданиеД1
Condition 273
Semicolon 3

```

```

ЗаданиеД2
[('Java', 381.0), ('C#', 253.0), ('Python', 173.0), ('JavaScript', 145.0), ('C++', 11.0), ('R', 3.0)]

```

```

ЗаданиеД3
{'C++': ['Condition', 'Loop', 'Shift', 'Add', 'Mul', 'Semicolon', 'Array'], 'C#': ['Condition', 'Loop', 'Shift', 'Add', 'Mul', 'Semicolon', 'Array']}

```