



**Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)**

**Факультет «ГУИМЦ»**

**Кафедра ИУ5 «Системы обработки информации и управления»**

Дисциплина «Базовые компоненты ИТ»

ОТЧЕТ

Рубежный контроль №1

Студент: Печуркин Д.С., группа ИУ5Ц-51Б

Преподаватель: Гапанюк Ю.Е.

2022г.

## Описание задания:

Вариант А, вариант предметной области №28.

1. «Кафедра» и «Студенческая группа» связаны соотношением один-ко-многим. Выведите список всех связанных кафедр и студенческих групп, отсортированный по кафедрам, сортировка по студенческим группам произвольная.
2. «Кафедра» и «Студенческая группа» связаны соотношением один-ко-многим. Выведите список кафедр с суммарным количеством студентов в каждой кафедре, отсортированных по суммарному количеству студентов.
3. «Кафедра» и «Студенческая группа» связаны соотношением многие-ко-многим. Выведите список всех кафедр, у которых в названии присутствует слово «ИУ», и список студенческих групп в них.

Класс «Кафедра», содержащий поля:

- id кафедры (id)
- название кафедры (name)

Класс «Студенческая группа», содержащий поля:

- id студенческой группы (id)
- название студенческой группы (name)
- кол-во студентов в студенческой группы (students)
- id кафедры, для реализации связи один-ко-многим (dep\_id)

Класс «Студенческие группы факультета» (для реализации связи один-ко-многим), содержащий поля:

- id кафедры (dep\_id)
- id студенческой группы (stud\_id)

## Листинг программы:

```
# Вариант 28 А Печуркин Д.С. ИУ5Ц-51Б
# from pprint import pprint
from operator import itemgetter

class Stud_Group:
    """Студенческая группа"""

    def __init__(self, id, name, count_students, dep_id):
        self.id = id
        self.name = name
        self.students = count_students
        self.dep_id = dep_id

class Departament:
    """Кафедра"""

    def __init__(self, id, name):
        self.id = id
        self.name = name

class Merge:
    """
    Студенческие группы кафедр
    для реализации связи многие-ко-многим
    """

    def __init__(self, dep_id, stud_id):
        self.dep_id = dep_id
        self.stud_id = stud_id

Stud_Groups = [
    Stud_Group(1, 'ИУ5Ц-51Б', 3, 1),
    Stud_Group(2, 'РК6-30', 10, 4),
    Stud_Group(3, 'ИБМ2-61', 6, 5),
    Stud_Group(4, 'ИБМ2-62', 6, 5),
    Stud_Group(5, 'МТ8-31', 8, 3),
    Stud_Group(6, 'ИУ3-31', 11, 2),
    Stud_Group(7, 'ИУ3-32', 11, 2),
    Stud_Group(8, 'ИУ5Ц-52Б', 5, 1),
    Stud_Group(9, 'ИУ5Ц-53Б', 7, 1),
]

Departaments = [
    Departament(1, 'ИУ5 - Системы обработки информации и управления'),
    Departament(2, 'ИУ3 - Информационные системы и телекоммуникации'),
    Departament(3, 'МТ8 - Материаловедение'),
    Departament(4, 'РК6 - Системы автоматизированного проектирования'),
    Departament(5, 'ИБМ2 - Экономика и организация производства'),
]

Merges = [
    Merge(1, 1),
    Merge(1, 8),
    Merge(1, 9),
    Merge(3, 5),
    Merge(5, 3),
    Merge(5, 4),
    Merge(2, 6),
    Merge(2, 7),
]

def main():
    # print('Вариант 28А Печуркин Д.С. ИУ5Ц-51Б\n')

    # Соединение данных один-ко-многим
```

```

one_to_many = [(group.name, group.students, dep.name)
               for dep in Departments
               for group in Stud_Groups
               if dep.id == group.dep_id]

print('Задание A1, сортировка по кафедрам по алфавиту\n')
res_0 = sorted(one_to_many, key=itemgetter(2))
print(res_0)

print('\nЗадание A2, суммарное количество студентов на каждой кафедре\n')
res_1 = []

for dep in Departments:
    # Список студентов кафедры
    deps = list(filter(lambda i: i[2] == dep.name, one_to_many))
    # Если кафедра не пустая
    if len(deps) > 0:
        # Кол-во студентов и суммарно студентов на кафедре
        stud_sum = sum([count for _, count, _ in deps])
        res_1.append((dep.name, stud_sum))

# Сортировка по количеству по убыванию
res_1 = sorted(res_1, key=itemgetter(1), reverse=True)
print(res_1)

print('\nЗадание A3, вывести список всех кафедр, название которых\n'
      '\t\t\tначинается на ИУ и список групп состоящих в них студентов\n')

res_2 = {}

many_to_many_temp = [(dep.name, merge.dep_id, merge.stud_id)
                     for dep in Departments
                     for merge in Merges
                     if dep.id == merge.dep_id]

many_to_many = [(stud.name, stud.students, name)
                 for name, dep_id, stud_id in many_to_many_temp
                 for stud in Stud_Groups if stud.id == stud_id]

# Перебираем все кафедры
for dep in Departments:
    if 'ИУ' in dep.name:
        # Список студентов кафедры
        list_studs = list(filter(lambda x: x[2] == dep.name, many_to_many))
        # Только названия кафедр
        list_deps = [name for name, _, _ in list_studs]
        # Добавляем результат в словарь
        # ключ - кафедра, значение - список групп
        res_2[dep.name] = list_deps

print(res_2)

if __name__ == '__main__':
    main()

```

## Результат выполнения программы:

Задание A1, сортировка по кафедрам по алфавиту

```

[('ИБМ2-61', 6, 'ИБМ2 - Экономика и организация производства'), ('ИБМ2-62',
6, 'ИБМ2 - Экономика и организация производства'), ('ИУ3-31', 11, 'ИУ3 -
Информационные системы и телекоммуникации'), ('ИУ3-32', 11, 'ИУ3 - Информационные
системы и телекоммуникации'), ('ИУ5Ц-51Б', 3, 'ИУ5 - Системы обработки информации
и управления'), ('ИУ5Ц-52Б', 5, 'ИУ5 - Системы обработки информации и

```

управления'), ('ИУ5Ц-53Б', 7, 'ИУ5 - Системы обработки информации и управления'), ('МТ8-31', 8, 'МТ8 - Материаловедение'), ('РК6-30', 10, 'РК6 - Системы автоматизированного проектирования')]

Задание А2, суммарное количество студентов на каждой кафедре

[('ИУ3 - Информационные системы и телекоммуникации', 22), ('ИУ5 - Системы обработки информации и управления', 15), ('ИБМ2 - Экономика и организация производства', 12), ('РК6 - Системы автоматизированного проектирования', 10), ('МТ8 - Материаловедение', 8)]

Задание А3, вывести список всех кафедр, название которых  
начинается на ИУ и список групп состоящих в них студентов

{'ИУ5 - Системы обработки информации и управления': ['ИУ5Ц-51Б', 'ИУ5Ц-52Б', 'ИУ5Ц-53Б'], 'ИУ3 - Информационные системы и телекоммуникации': ['ИУ3-31', 'ИУ3-32']}