

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
им. Н.Э. Баумана

Факультет “Информатика и системы управления”
Кафедра “Системы обработки информации и управления”



Дисциплина “Парадигмы и конструкции языков программирования”

Отчет по рубежному контролю №1
“Основные конструкции языка Python ”

Выполнил:
Студент группы ИУ5-34Б
Савинский А. Ю.
Преподаватель:
Нардид А.Н.

Москва 2025

Код программы

```
from typing import List, Tuple

class University:
    def __init__(self, id: int, name: str):
        self.id = id
        self.name = name

    def __repr__(self):
        return f"University(id={self.id}, name={self.name!r})"

class Faculty:
    def __init__(self, id: int, name: str, students: int, university_id: int):
        self.id = id
        self.name = name
        self.students = students
        self.university_id = university_id

    def __repr__(self):
        return (f"Faculty(id={self.id}, name={self.name!r}, "
                f"students={self.students}, university_id={self.university_id})")

class FacultyInUniversity:
    def __init__(self, faculty_id: int, university_id: int):
        self.faculty_id = faculty_id
        self.university_id = university_id

    def __repr__(self):
        return f"FacultyInUniversity(faculty_id={self.faculty_id}, "
                f"university_id={self.university_id})"

universities: List[University] = [
    University(1, "МГТУ им. Н. Э. Баумана"),
    University(2, "ВШЭ"),
    University(3, "МФТИ"),
]

faculties: List[Faculty] = [
    Faculty(1, "Информатика и системы управления", 1800, 1),
    Faculty(2, "Компьютерные науки", 1500, 2),
    Faculty(3, "Прикладная математика и информатика", 1600, 3),
    Faculty(4, "Физика", 1400, 3),
    Faculty(5, "Радиотехника", 900, 1),
    Faculty(6, "Инженерный бизнес и менеджмент", 1200, 1),
]

fac_in_uni: List[FacultyInUniversity] = [
    FacultyInUniversity(4, 3),
    FacultyInUniversity(4, 2),
    FacultyInUniversity(5, 1),
    FacultyInUniversity(5, 3),
    FacultyInUniversity(1, 1),
    FacultyInUniversity(2, 2),
    FacultyInUniversity(3, 3),
    FacultyInUniversity(6, 1),
]

uni_by_id = {u.id: u for u in universities}
fac_by_id = {f.id: f for f in faculties}

# 1) один-ко-многим: список всех факультетов, у которых название начинается с «И»
def query_1_faculties_starting_with_i(faculties: List[Faculty]) -> List[Tuple[str, str]]:
    pairs = [
        (f.name, uni_by_id[f.university_id].name)
        for f in faculties
        if f.name.upper().startswith("И")
    ]
    return sorted(pairs, key=lambda x: (x[0], x[1]))

# 2) один-ко-многим: список университетов с минимальной численностью студентов на факультете,
# отсортированный по этому минимуму
```

```

def query_2_min_students_per_university(faculties: List[Faculty]) -> List[Tuple[str,
int]]:
    groups = {}
    for f in faculties:
        groups.setdefault(f.university_id, []).append(f.students)

    agg = [(uni_by_id[uid].name, min(studs)) for uid, studs in groups.items()]
    return sorted(agg, key=lambda x: x[1])

# 3) многие-ко-многим: список всех связанных факультетов и университетов,
# отсортировано по названию факультетов
def query_3_all_fac_uni_m2m(links: List[FacultyInUniversity]) -> List[Tuple[str,
str]]:
    pairs = [(fac_by_id[link.faculty_id].name, uni_by_id[link.university_id].name) for
link in links]
    return sorted(pairs, key=lambda x: x[0])

def main():
    q1 = query_1_faculties_starting_with_i(faculties)
    q2 = query_2_min_students_per_university(faculties)
    q3 = query_3_all_fac_uni_m2m(fac_in_uni)

    print("---- Задание B1 ----\n Факультеты, начинающиеся на «И»:")
    for name_f, name_u in q1:
        print(f"    - {name_f} - {name_u}")

    print("\n---- Задание B2 ----\n Университеты с минимальной численностью студентов
на факультете, по возрастанию минимума:")
    for name_u, min_studs in q2:
        print(f"    - {name_u}: {min_studs} студ.")

    print("\n---- Задание B3 ----\n Связанные факультеты и университеты, отсортировано
по факультетам:")
    for name_f, name_u in q3:
        print(f"    - {name_f} - {name_u}")

if __name__ == "__main__":
    main()

```

Результат работы программы

```

● xоqous1kkke@MacBook-Pro-Andrej-3 Savinskii_PCPL_Labs % python3 rk1/rk1.py
---- Задание B1 ----
Факультеты, начинающиеся на «И»:
- Инженерный бизнес и менеджмент – МГТУ им. Н. Э. Баумана
- Информатика и системы управления – МГТУ им. Н. Э. Баумана

---- Задание B2 ----
Университеты с минимальной численностью студентов на факультете, по возрастанию минимума:
- МГТУ им. Н. Э. Баумана: 900 студ.
- МФТИ: 1400 студ.
- ВШЭ: 1500 студ.

---- Задание B3 ----
Связанные факультеты и университеты, отсортировано по факультетам:
- Инженерный бизнес и менеджмент – МГТУ им. Н. Э. Баумана
- Информатика и системы управления – МГТУ им. Н. Э. Баумана
- Компьютерные науки – ВШЭ
- Прикладная математика и информатика – МФТИ
- Радиотехника – МГТУ им. Н. Э. Баумана
- Радиотехника – МФТИ
- Физика – МФТИ
- Физика – ВШЭ
○ xоqous1kkke@MacBook-Pro-Andrej-3 Savinskii_PCPL_Labs % 

```