```
from operator import itemgetter


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


class StudentGroup:
    def __init__(self, id, number, department_id, student_count):
        self.id = id
        self.number = number
        self.department_id = department_id
        self.student_count = student_count


class StudentsInDepartment:
    def __init__(self, department_id, student_id):
        self.department_id = department_id
        self.student_id = student_id


def get_related_students_and_departments(student_groups, departments,
students_in_departments):
    one_to_many = [(group.number, department.name) for group in student_groups
for department in departments if
                   group.department_id == department.id]
    return sorted(one_to_many, key=itemgetter(1))


def get_department_student_counts(student_groups, departments,
students_in_departments):
    res_2_unsorted = []
    for department in departments:
```

```
        group_student_counts = [group.student_count for group in student_groups if group.department_id == department.id]
        total_student_count = sum(group_student_counts)
        res_2_unsorted.append((department.name, total_student_count))
    return sorted(res_2_unsorted, key=itemgetter(1), reverse=True)

def get_group_student_counts(student_groups):
    res_3 = {}
    for group in student_groups:
        res_3[group.number] = group.student_count
    return res_3

def main():
    # ... (remaining code for data initialization)

    print('Запрос 1: Список связанных студентов и кафедр, отсортированный по кафедрам')
    res_1 = get_related_students_and_departments(student_groups, departments, students_in_departments)
    print(res_1)

    print('\nЗапрос 2: Список кафедр с суммарным количеством студентов в каждой группе')
    res_2 = get_department_student_counts(student_groups, departments, students_in_departments)
    print(res_2)

    print('\nЗапрос 3: Список студенческих групп и количества студентов')
    res_3 = get_group_student_counts(student_groups)
    for key, value in res_3.items():
```

```
        print(key, value)


if __name__ == '__main__':
    main()
```