

Рубежный контроль №1
Вариант

32	Колонка данных	Таблица данных
----	----------------	----------------

Вариант Г.

1. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список всех отделов, у которых название начинается с буквы «А», и список работающих в них сотрудников.
2. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список отделов с максимальной зарплатой сотрудников в каждом отделе, отсортированный по максимальной зарплате.
3. «Отдел» и «Сотрудник» связаны соотношением многие-ко-многим. Выведите список всех связанных сотрудников и отделов, отсортированный по отделам, сортировка по сотрудникам произвольная.

Текст программы

```
from operator import itemgetter
class DataColumn:
    def __init__(self, id, text_data, size, table_id):
        self.id = id
        self.text_data = text_data
        self.size = size
        self.table_id = table_id

class DataTable:
    def __init__(self, id, table_name):
        self.id = id
        self.table_name = table_name

class TableColumns:
    def __init__(self, column_id, table_id):
        self.column_id = column_id
        self.table_id = table_id

tables = [DataTable(1, "Clients"),
          DataTable(2, "Transactions"),
          DataTable(3, "Shipments"),
          DataTable(4, "Stocks"),
```


#Г1

```
g1_temp = list(filter(lambda i: i[2][0] == 'S', one_to_many))
```

```
g1_t = list(set(list(x[2] for x in g1_temp)))
```

```
print(g1_t)
```

```
g1_c = list(x[0] for x in g1_temp)
```

```
print(g1_c)
```

#Г2

```
g2_temp1 = list((x[2], x[1]) for x in one_to_many)
```

```
g2_temp2 = list(reversed(sorted(list(set(list(filter(lambda i: i[1] == max(list(x[1] for x in  
one_to_many if x[2] == i[0])), g2_temp1)))), key=itemgetter(1))))
```

```
print(g2_temp2)
```

#Г3

```
print(sorted(list((x[0], x[2]) for x in many_to_many), key=itemgetter(1)))
```

```
if __name__ == '__main__':
```

```
    main()
```