

**О Т Ч Е Т**  
**по рубежному контролю № 1**

## Вариант 25

Москва, 2021

## Описание задания

1. «Раздел» и «Документ» связаны соотношением один-ко-многим. Выведите список всех связанных разделов и отделов, отсортированный по разделам, сортировка по документам произвольная.
2. «Раздел» и «Документ» связаны соотношением один-ко-многим. Выведите список разделов с количеством страниц в каждом разделе, отсортированный по страницам документа.
3. «Раздел» и «Документ» связаны соотношением многие-ко-многим. Выведите список всех документов, к которым относиться конкретный раздел.

Класс «Раздел» содержит поля:

- id раздела (id)
- название раздела (name\_p)

Класс «Документ» содержит поля:

- id документа (id)
- название документа (name)
- количество страниц в документе (page\_num)
- id раздела, для реализации связи многие ко многим (part\_id)

Класс «Документы в разделах»:

- id документа (part\_id)
- id раздела (docum\_id)

## Код программы

""""ПК №1, Свинцов Артемий ИУ5Ц-52Б

Вариант Б, вариант предметной области 25""""

```
from operator import itemgetter
```

```
class part:
```

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

```
        self.id = id
```

```
        self.name_p = name_p
```

```
class docum:

    def __init__(self, id, name, page_num, part_id):

        self.id = id

        self.name = name

        self.page_num = page_num

        self.part_id = part_id
```

```
class part_doc:

    def __init__(self, docum_id, part_id):

        self.docum_id = docum_id

        self.part_id = part_id
```

```
parts = [

    part(1, 'Кодекс'),

    part(2, 'Устав организации'),

    part(3, 'Федеральный закон'),

    part(4, 'Нормативный акт'),

]
```

```
docums = [

    docum(1, 'Уголовный', 15, 1),

    docum(2, 'Об образовании', 20, 3),

    docum(3, 'Организация труда', 10, 2),

    docum(4, 'Запрет продажи алкоголя', 31, 4),

    docum(5, 'О внутреннем распорядке', 16, 2),

]
```

```
part_docs = [  
    part_doc(1, 1),  
    part_doc(2, 3),  
    part_doc(3, 2),  
    part_doc(4, 1),  
    part_doc(5, 2),  
]
```

```
def main():  
    one_to_many = [(g.name, g.page_num, c.name_p)  
        for g in docums  
        for c in parts  
        if g.part_id == c.id]  
  
    many_to_many_temp = [(c.name_p, gc_s.part_id, gc_s.docum_id)  
        for c in parts  
        for gc_s in part_docs  
        if c.id == gc_s.part_id]  
  
    many_to_many = [(g.name, g.page_num, part_name)  
        for part_name, part_id, docum_id in many_to_many_temp  
        for g in docums if g.id == docum_id]  
  
    print("Задание Б1")  
  
    res_11 = sorted(one_to_many, key=itemgetter(0))  
  
    print(res_11)  
  
    print("\nЗадание Б2")
```

```

res_12_unsorted = []

for c in parts:

    parts_s = list(filter(lambda i: i[2]==c.name_p, one_to_many))

    if len(parts_s) > 0:

        parts_amount = [page_num for _,page_num,_ in parts_s]

        parts_amount_sum = sum(parts_amount)

        res_12_unsorted.append((c.name_p, parts_amount_sum))


res_12 = sorted(res_12_unsorted, key=itemgetter(1), reverse=True)

print(res_12)

print("\nЗадание Б3")

res_13 = {}

for c in parts:

    if len(c.name_p) > 0:

        part_s = list(filter(lambda i: i[2]==c.name_p, many_to_many))

        part_s_name = [x for x,_,_ in part_s]

        res_13[c.name_p] = part_s_name

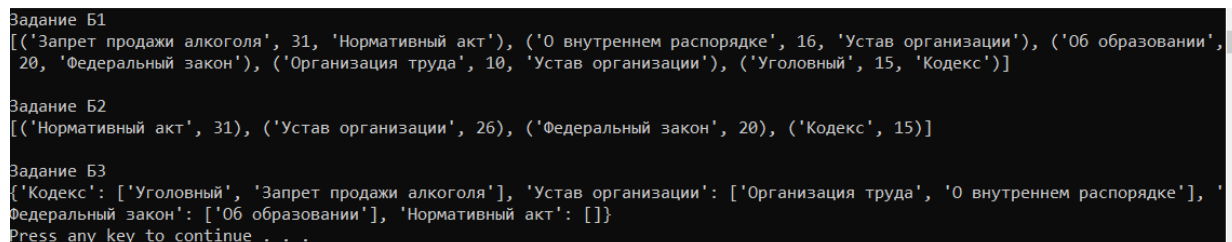
print(res_13)

if __name__ == '__main__':

    main()

```

## Результат выполнения программы



```

Задание Б1
[('Запрет продажи алкоголя', 31, 'Нормативный акт'), ('О внутреннем распорядке', 16, 'Устав организации'), ('Об образовании', 20, 'Федеральный закон'), ('Организация труда', 10, 'Устав организации'), ('Уголовный', 15, 'Кодекс')]

Задание Б2
[('Нормативный акт', 31), ('Устав организации', 26), ('Федеральный закон', 20), ('Кодекс', 15)]

Задание Б3
{'Кодекс': ['Уголовный', 'Запрет продажи алкоголя'], 'Устав организации': ['Организация труда', 'О внутреннем распорядке'], 'Федеральный закон': ['Об образовании'], 'Нормативный акт': []}
Press any key to continue . . .

```