

Шилина А.Ю. ИУ5-31Б;
ВАРИАНТ 25. Раздел. Документ;
Вариант А.

Код программы:

```
from operator import itemgetter
class Document:
    """документы"""

    def __init__(self, id, title, pages, section_id):
        self.id = id
        self.title = title
        self.pages = pages
        self.section_id = section_id
class Section:
    """разделы"""
    def __init__(self, id, name):
        self.id = id
        self.name = name
class DocSection:
    """документы раздела м к м"""
    def __init__(self, section_id, doc_id):
        self.section_id = section_id
        self.doc_id = doc_id

# тестовые данные для разделов
sections = [
    Section(1, 'Исторический раздел'),
    Section(2, 'Научный отдел'),
    Section(3, 'Литературный раздел'),
    Section(4, 'Философский отдел'),
    Section(5, 'Раздел математики'),
]
# тестовые данные для документов
documents = [
    Document(1, 'История Древнего Мира', 300, 1),
    Document(2, 'Квантовая механика', 250, 2),
    Document(3, 'Сборник стихов Лермонтова', 200, 3),
    Document(4, 'Метафизика Аристотеля', 150, 4),
    Document(5, 'Геометрия Евклида', 350, 5),
]
# тестовые данные для связей
docs_sections = [
    DocSection(1, 1), # Исторический раздел - История Древнего Мира
    DocSection(2, 2), # Научный отдел - Квантовая механика
    DocSection(3, 3), # Литературный раздел - Сборник стихов Лермонтова
    DocSection(4, 4), # Философский отдел - Метафизика Аристотеля
    DocSection(5, 5), # Раздел математики - Геометрия Евклида
    DocSection(1, 3), # Исторический раздел - Сборник стихов Лермонтова //
НЕ ЗАБЫТЬ ПРО М К М
    DocSection(2, 5), # Научный отдел - Геометрия Евклида //НЕ ЗАБЫТЬ ПРО М
К М
]
def main():
    # 1 к м
    one_to_many = [(d.title, d.pages, s.name)
                    for s in sections
                    for d in documents
                    if d.section_id == s.id]

    # м к м
    many_to_many_temp = [(s.name, ds.section_id, ds.doc_id)
                          for s in sections
```

```

        for ds in docs_sections
        if s.id == ds.section_id]

many_to_many = [(d.title, d.pages, section_name)
                 for section_name, section_id, doc_id in many_to_many_temp
                 for d in documents if d.id == doc_id]

# ПУНКТ A1
# список всех связанных док-ов и разделов, сорт по разделам
print('Задание A1')
res_11 = sorted(one_to_many, key=itemgetter(2))
print(res_11)

# ПУНКТ A2
# список разделов с сум. кол-вом страниц док-ов, сорт по сум. странице
print('\nЗадание A2')
res_12_unsorted = []
for s in sections:
    # список документов раздела
    s_docs = list(filter(lambda i: i[2] == s.name, one_to_many))
    if len(s_docs) > 0:
        s_pages = [pages for _, pages, _ in s_docs]
        s_pages_sum = sum(s_pages)
        res_12_unsorted.append((s.name, s_pages_sum))
# сортировка
res_12 = sorted(res_12_unsorted, key=itemgetter(1), reverse=True)
print(res_12)

# ПУНКТ A3
# "раздел" в названии и доки в них
print('\nЗадание A3')
res_13 = {}
for s in sections:
    if 'раздел' in s.name.lower():
        s_docs = list(filter(lambda i: i[2] == s.name, many_to_many))
        s_docs_titles = [x for x, _, _ in s_docs]
        res_13[s.name] = s_docs_titles
print(res_13)
if __name__ == '__main__':
    main()

```

Результат:

Задание A1

[('История Древнего Мира', 300, 'Исторический раздел'), ('Сборник стихов Лермонтова', 200, 'Литературный раздел'), ('Квантовая механика', 250, 'Научный отдел'), ('Геометрия Евклида', 350, 'Раздел математики'), ('Метафизика Аристотеля', 150, 'Философский отдел')]

Задание A2

[('Раздел математики', 350), ('Исторический раздел', 300), ('Научный отдел', 250), ('Литературный раздел', 200), ('Философский отдел', 150)]

Задание A3

{'Исторический раздел': ['История Древнего Мира', 'Сборник стихов Лермонтова'], 'Литературный раздел': ['Сборник стихов Лермонтова'], 'Раздел математики': ['Геометрия Евклида']}