Студент: Пустовалов Григорий

Группа: ИУ5-32Б

Вариант: 20Б

Условие:

1. «Деталь» и «Поставщик» связаны соотношением один-ко-многим. Выведите список всех связанных поставщиков и деталей, отсортированный по поставщикам, сортировка по деталям произвольная.

2. «Деталь» и «Поставщик» связаны соотношением один-ко-многим. Выведите список деталей с количеством поставщиков у каждой детали, отсортированный по количеству поставщиков.

3. «Деталь» и «Поставщик» связаны соотношением многие-ко-многим. Выведите список всех поставщиков, у которых фамилия заканчивается на «ов», и названия их деталей.

Код программы:

файл provider.py

```
class Provider:
    """Поставщик"""
    def __init__(self, id, fio, sal, det_id):
        self.id = id
        self.fio = fio
        self.sal = sal
        self.det_id = det_id
```

файл detail.py

```
class Detail:
    """Деталь"""

    def __init__(self, id, name):
        self.id = id
        self.name = name
```

файл detPro.py

```
class DetPro:
    """
    'Поставка детали' для реализации
    связи многие-ко-многим
    """
    def __init__(self, det_id, pro_id):
        self.det_id = det_id
        self.pro_id = pro_id
```

файл main.py

```
from operator import itemgetter
from detail import Detail
from provider import Provider
from detPro import DetPro
from arrays import dets
from arrays import providers
from arrays import prov_dets
```

```python
def main():
    """Основная функция"""
    # Соединение данных многие-ко-многим
    many_to_many_temp = [(d.name, ed.det_id, ed.pro_id)
                         for d in dets
                         for ed in prov_dets
                         if d.id == ed.pro_id
                         ]

    many_to_many = [(e.fio, e.sal, dep_name)
                    for dep_name, det_id, pro_id in many_to_many_temp
                    for e in providers
                    if e.id == pro_id
                    ]

    # Соединение данных один-ко-многим
    one_to_many = [(e.fio, e.sal, e.det_id, d.name)
                   for d in dets
                   for e in providers
                   if e.det_id == d.id
                   ]

    res_1 = sorted(one_to_many, key=itemgetter(0))

    print('Задание Б1')
    print('№|Фамилия|ЗП|название детали')
    print('_____________________________________________')
    for num, i in enumerate(res_1):
        print("{}|{}|{}|{}|{}".format(num + 1, i[0], i[1], i[2], i[3]))


    res_2 = []

    for i in dets:
        detailsLambda = list(filter(lambda j: j[3] == i.name, one_to_many))
        if len(detailsLambda) > 0:
            res_2.append((i.name, len(detailsLambda)))
    res_2 = sorted(res_2, key = itemgetter(1), reverse=True)

    print('Задание Б2')
    print('№|Название детали| количество поставщиков')
    print('_____________________________________________')
    for num, i in enumerate(res_2):
        print("{}|{}|{}".format(num + 1, i[0], i[1]))

    res_3 = []
    for i in many_to_many:
        if (i[0][-2] == "о") and (i[0][-1] == "в"):
            print(i)
            res_3.append(i)

    print('Задание Б3')
    print('№|Поставщик| название детали')
    print('_____________________________________________')
    for num, i in enumerate(res_3):
        print("{}|{}|{}".format(num + 1, i[0], i[2]))


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

Результат выполнения программы:

```
C:\Users\Григорий\PycharmProjects\RK1_3se
Задание Б1
№|Фамилия|ЗП|название детали
----------------------------------------
1|Васькин|10000|1|вал
2|Ефимов|95000|3|корпус
3|Иванов|35000|3|корпус
4|Павлов|999|2|гайка
5|Петров|55000|2|гайка
6|Сидоров|25000|3|корпус
Задание Б2
№|Название детали| количество поставщиков
----------------------------------------
1|корпус|3
2|гайка|2
3|вал|1
Задание Б3
№|Поставщик| название детали
----------------------------------------
```

```
----------------------------------------
1|корпус|3
2|гайка|2
3|вал|1
Задание Б3
№|Поставщик| название детали
----------------------------------------
1|Сидоров|вал
2|Сидоров|вал
3|Сидоров|вал
4|Петров|гайка
5|Иванов|вал (другой)
6|Ефимов|гайка (другой)
7|Ефимов|гайка (другой)

Process finished with exit code 0
```