

Отчёт РК1 по дисциплине “Парадигмы и конструкции языков программирования”

Вариант запросов

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.

№ варианта	Класс 1	Класс 2
4	Компьютер	Дисплейный класс

Код программы:

```
class Computer:
    def __init__(self, comp_id, name, price):
        self.comp_id = comp_id
        self.name = name
        self.price = price

class DisplayClass:
    def __init__(self, display_id, name):
        self.display_id = display_id
        self.name = name
```

```

class ComputerDisplay:
    def __init__(self, comp_id, display_id):
        self.comp_id = comp_id
        self.display_id = display_id

# Создание тестовых данных
computers = [
    Computer(1, 'Dell XPS', 1200),
    Computer(2, 'Apple MacBook', 2400),
    Computer(3, 'Lenovo ThinkPad', 900),
    Computer(4, 'Asus ROG', 1500),
]

display_classes = [
    DisplayClass(1, 'Advanced Display Technology'),
    DisplayClass(2, 'Basic Display Package'),
    DisplayClass(3, 'Advanced Display Tech'),
]

computer_displays = [
    ComputerDisplay(1, 1),
    ComputerDisplay(2, 2),
    ComputerDisplay(3, 1),
    ComputerDisplay(4, 3),
    ComputerDisplay(3, 2),
]

# Запрос 1
print("Компьютеры с дисплейным классом, начинающимся на 'A':")
print([(d.name, [c.name for c in computers if c.comp_id in [cd.comp_id for cd in computer_displays if cd.display_id == d.display_id]]
        for d in display_classes if d.name.startswith('A')])])

# Запрос 2
print("\nДисплейные классы с максимальной ценой компьютера:")
max_price_by_display = {d.display_id: max([c.price for c in computers if
c.comp_id in [cd.comp_id for cd in computer_displays if cd.display_id ==
d.display_id]], default=0)
                        for d in display_classes}
sorted_by_max_price = sorted(max_price_by_display.items(), key=lambda x:
x[1], reverse=True)

for display_id, price in sorted_by_max_price:
    display_name = next((d.name for d in display_classes if d.display_id ==
display_id), "Unknown")
    print(f"{display_name}: {price}")

# Запрос 3
print("\nВсе связанные компьютеры и дисплейные классы:")
related_data = [(f"Компьютер: {c.name}", f"Дисплейный класс: {d.name}")
                 for cd in computer_displays
                 for c in computers
                 for d in display_classes
                 if cd.comp_id == c.comp_id and cd.display_id == d.display_id]

# Сортировка по дисплейному классу
related_data.sort(key=lambda x: x[1])

for data in related_data:
    print(f"{data[1]} - {data[0]}")

```

## Результат выполнения программы

```
C:\Users\fenix\PycharmProjects\RK1\.venv\bin\python.exe C:\Users\fenix\PycharmProjects\RK1\main.py
Компьютеры с дисплейным классом, начинающимся на 'A':
[('Advanced Display Technology', ['Dell XPS', 'Lenovo ThinkPad']), ('Advanced Display Tech', ['Asus ROG'])]

Дисплейные классы с максимальной ценой компьютера:
Basic Display Package: 2400
Advanced Display Tech: 1500
Advanced Display Technology: 1200

Все связанные компьютеры и дисплейные классы:
Дисплейный класс: Advanced Display Tech - Компьютер: Asus ROG
Дисплейный класс: Advanced Display Technology - Компьютер: Dell XPS
Дисплейный класс: Advanced Display Technology - Компьютер: Lenovo ThinkPad
Дисплейный класс: Basic Display Package - Компьютер: Apple MacBook
Дисплейный класс: Basic Display Package - Компьютер: Lenovo ThinkPad

Process finished with exit code 0
```