

```

1  from operator import itemgetter
2
3  class Computer:
4      """Компьютер"""
5      def __init__(self, id, model, price, display_class_id):
6          self.id = id
7          self.model = model
8          self.price = price
9          self.display_class_id = display_class_id
10
11  class DisplayClass:
12      """Дисплейный класс"""
13      def __init__(self, id, name):
14          self.id = id
15          self.name = name
16
17  class ComputerDisplayClass:
18      """Компьютеры в дисплейных классах для связи многие-ко-многим"""
19      def __init__(self, display_class_id, computer_id):
20          self.display_class_id = display_class_id
21          self.computer_id = computer_id
22
23  # Дисплейные классы
24  display_classes = [
25      DisplayClass(1, 'А-101'),
26      DisplayClass(2, 'Б-202'),
27      DisplayClass(3, 'А-303'),
28      DisplayClass(4, 'Б-404'),
29      DisplayClass(5, 'А-505'),
30  ]
31
32  # Компьютеры
33  computers = [
34      Computer(1, 'Dell Optiplex', 50000, 1),      # А-101
35      Computer(2, 'HP ProDesk', 45000, 2),      # Б-202
36      Computer(3, 'Lenovo ThinkCentre', 60000, 3), # А-303
37      Computer(4, 'Acer Veriton', 40000, 3),      # А-303
38      Computer(5, 'Asus ExpertCenter', 55000, 3), # А-303
39      Computer(6, 'Fujitsu Esprimo', 48000, 5),    # А-505
40      Computer(7, 'MSI Pro', 52000, 5),          # А-505
41  ]
42
43  # Связь многие-ко-многим (дополнительные связи)
44  computers_display_classes = [
45      ComputerDisplayClass(1, 3), # Lenovo в А-101 (дополнительно)
46      ComputerDisplayClass(2, 5), # Asus в Б-202 (дополнительно)
47  ]
48

```

```

49 def main():
50     """Основная функция"""
51
52     print('Задание Г1')
53     print('Список всех дисплейных классов, у которых название начинается буквой «А», и список компьютеров в них:')
54
55     # ЗАДАНИЕ Г1: связь один-ко-многим
56     one_to_many = [(c, dc) for dc in display_classes for c in computers if c.display_class_id == dc.id]
57
58     result_1 = {}
59     for computer, display_class in one_to_many:
60         if display_class.name.startswith('А'):
61             if display_class.name not in result_1:
62                 result_1[display_class.name] = []
63             result_1[display_class.name].append(computer)
64
65     for class_name, comp_list in sorted(result_1.items()):
66         print(f'\n{class_name}:')
67         for comp in comp_list:
68             print(f"    {comp.model} (цена: {comp.price} руб.)")
69
70     print('\n' + '='*60)
71     print('Задание Г2')
72     print('Список дисплейных классов максимальной ценой компьютеров в каждом классе, отсортированный по максимальной цене:')
73
74     # ЗАДАНИЕ Г2: связь один-ко-многим
75     class_computers = {}
76     for computer, display_class in one_to_many:
77         if display_class.name not in class_computers:
78             class_computers[display_class.name] = []
79         class_computers[display_class.name].append(computer.price)
80
81     max_prices = {}
82     for class_name, prices in class_computers.items():
83         max_prices[class_name] = max(prices)
84
85     result_2 = sorted(max_prices.items(), key=itemgetter(1), reverse=True)
86
87     for class_name, max_price in result_2:
88         print(f'{class_name}: {max_price} руб.')
89
90     print('\n' + '='*60)
91     print('Задание Г3')
92     print('Список всех связанных компьютеров и дисплейных классов, отсортированный по классам:')
93
94     # ЗАДАНИЕ Г3: ВСЕ связи (один-ко-многим + многие-ко-многим)
95     all_connections = []
96

```

```

94 # ЗАДАНИЕ Г3: ВСЕ СВЯЗИ (ОДИН-КО-МНОГИМ + МНОГИЕ-КО-МНОГИМ)
95 all_connections = []
96
97 # 1. Добавляем все связи один-ко-многим
98 for computer, display_class in one_to_many:
99     all_connections.append((computer, display_class))
100
101 # 2. Добавляем все связи многие-ко-многим
102 for cdc in computers_display_classes:
103     computer = next((c for c in computers if c.id == cdc.computer_id), None)
104     display_class = next((dc for dc in display_classes if dc.id == cdc.display_class_id), None)
105     if computer and display_class:
106         # Проверяем, нет ли уже такой связи в all_connections
107         connection_exists = any(
108             conn[0].id == computer.id and conn[1].id == display_class.id
109             for conn in all_connections
110         )
111         if not connection_exists:
112             all_connections.append((computer, display_class))
113
114 # Сортируем по названию класса (основная сортировка)
115 all_connections.sort(key=lambda x: x[1].name)
116
117 # Группируем по классам
118 result_3 = {}
119 for computer, display_class in all_connections:
120     if display_class.name not in result_3:
121         result_3[display_class.name] = []
122     result_3[display_class.name].append(computer)
123
124 # Для каждого класса сортируем компьютеры произвольно (по убыванию цены)
125 for class_name, comp_list in sorted(result_3.items()):
126     comp_list_sorted = sorted(comp_list, key=lambda x: x.price, reverse=True)
127     print(f"\n{class_name}:")
128     for comp in comp_list_sorted:
129         print(f"    - {comp.model} (цена: {comp.price} руб.)")
130
131 if __name__ == '__main__':
132     main()
133

```

```
kirill@WIN-OHQ18823BK3:~/Programming_Paradigms/RK1$ python3 main.py
Задание Г1
Список всех дисплейных классов, у которых название начинается с буквы «А», и список компьютеров в них:

А-101:
- Dell Optiplex (цена: 50000 руб.)

А-303:
- Lenovo ThinkCentre (цена: 60000 руб.)
- Acer Veriton (цена: 40000 руб.)
- Asus ExpertCenter (цена: 55000 руб.)

А-505:
- Fujitsu Esprimo (цена: 48000 руб.)
- MSI Pro (цена: 52000 руб.)

=====
Задание Г2
Список дисплейных классов с максимальной ценой компьютеров в каждом классе, отсортированный по максимальной цене:
А-303: 60000 руб.
А-505: 52000 руб.
А-101: 50000 руб.
Б-202: 45000 руб.

=====
Задание Г3
Список всех связанных компьютеров и дисплейных классов, отсортированный по классам:

А-101:
- Lenovo ThinkCentre (цена: 60000 руб.)
- Dell Optiplex (цена: 50000 руб.)

А-303:
- Lenovo ThinkCentre (цена: 60000 руб.)
- Asus ExpertCenter (цена: 55000 руб.)
- Acer Veriton (цена: 40000 руб.)

А-505:
- MSI Pro (цена: 52000 руб.)
- Fujitsu Esprimo (цена: 48000 руб.)

Б-202:
- Asus ExpertCenter (цена: 55000 руб.)
- HP ProDesk (цена: 45000 руб.)
```