



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

Факультет «Информатика и системы управления»
Кафедра «Системы обработки информации и управления»

Рубежный контроль № 1
по курсу “Разработка интернет-приложений”

Выполнил:
студент группы ИУ5-53Б
Волгина А. Д.
17.10.21

Проверил:
Гапанюк Ю.Е.

2021 г.

Вариант предметной области (8):

Класс 1 – жёсткий диск, класс 2 – компьютер.

Вариант запросов (B):

1. «Жёсткий диск» и «Компьютер» связаны соотношением один-ко-многим. Выведите список всех жёстких дисков с ёмкостью >100 ГБ и ID их компьютеров.
2. «Жёсткий диск» и «Компьютер» связаны соотношением один-ко-многим. Выведите список компьютеров с минимальной занятостью жёстких дисков в каждом компьютере, отсортированный по минимальной занятости.
3. «Жёсткий диск» и «Компьютер» связаны соотношением многие-ко-многим. Выведите список всех связанных жёстких дисков и компьютеров, отсортированный по жёстким дискам, сортировка по компьютерам произвольная.

Текст программы:

```
class Computer:
```

```
    def __init__(self, ID, company, price):
```

```
        self.ID = ID
```

```
        self.company = company
```

```
        self.price = price
```

```
class HardDrive:
```

```
    def __init__(self, ID, name, ComputerID, capacity, filled):
```

```
        self.ID = ID
```

```
        self.name = name
```

```
        self.ComputerID = ComputerID
```

```
        self.capacity = capacity
```

```
        self.filled = filled
```

```
class ComputerHardDrives:
```

```
def __init__(self, HardID, ComputerID):  
    self.HardID = HardID,  
    self.ComputerID = ComputerID
```

```
def main():
```

```
    HardDrives = [HardDrive(1, 'A', 1, 128, 77),  
                   HardDrive(2, 'B', 1, 150, 100),  
                   HardDrive(3, 'C', 1, 80, 30),  
                   HardDrive(4, 'D', 2, 128, 64),  
                   HardDrive(5, 'E', 2, 64, 60),  
                   HardDrive(6, 'A', 3, 210, 100),  
                   HardDrive(7, 'B', 3, 180, 40),  
                   HardDrive(8, 'A', 4, 80, 70),  
                   HardDrive(9, 'B', 4, 64, 40)  
    ]
```

```
    Computers = [Computer(1, 'HP', 40),  
                 Computer(2, 'Dell', 50),  
                 Computer(3, 'HP', 45),  
                 Computer(4, 'Apple', 60)  
    ]
```

```
    HardsOfComps = [ComputerHardDrives(1, 1),  
                    ComputerHardDrives(1, 3),  
                    ComputerHardDrives(2, 1),  
                    ComputerHardDrives(3, 1),
```

```

        ComputerHardDrives(3, 3),
        ComputerHardDrives(3, 4),
        ComputerHardDrives(4, 2),
        ComputerHardDrives(5, 2),
        ComputerHardDrives(5, 4),
        ComputerHardDrives(6, 3),
        ComputerHardDrives(7, 3),
        ComputerHardDrives(7, 1),
        ComputerHardDrives(7, 4),
        ComputerHardDrives(8, 4),
        ComputerHardDrives(9, 4)
    ]

    print("Task 1")

    for drive in HardDrives:

        if(drive.capacity > 100):

            print("ID диска: {0}\nЁмкость диска: {1}\nID компьютера: {2}".format(drive.ID, drive.capacity, drive.ComputerID))

    print("\nTask 2")

    MinFill = {}

    for comp in Computers:

        MinFill[str(comp.ID)] = "1000"

    for drive in HardDrives:

        if(drive.filled < int(MinFill[str(drive.ComputerID)])):

            MinFill[str(drive.ComputerID)] = drive.filled

    sorted_values = sorted(MinFill.values())

```

```

sorted_minfill = { }

keys = list(MinFill.keys())

for value in sorted_values:

    for key in keys:

        if(MinFill[key] == value):

            sorted_minfill[key] = value

            keys.remove(key)

            print("ID компьютера: {0}\nМинимальная занятость диска:
{1}".format(key, value))

            break

print("\nTask 3")

for drive in HardDrives:

    for comp in HardsOfComps:

        if(comp.HardID[0] == drive.ID):

            print("Жёсткий диск: {0}\nКомпьютер: {1}".format(drive.ID,
comp.ComputerID))

if __name__ == "__main__":

    main()

```

Результат выполнения:

Task 1

ID диска: 1
Ёмкость диска: 128
ID компьютера: 1
ID диска: 2
Ёмкость диска: 150
ID компьютера: 1
ID диска: 4
Ёмкость диска: 128
ID компьютера: 2
ID диска: 6
Ёмкость диска: 210
ID компьютера: 3
ID диска: 7
Ёмкость диска: 180
ID компьютера: 3

Task 2

ID компьютера: 1
Минимальная занятость диска: 30
ID компьютера: 3
Минимальная занятость диска: 40
ID компьютера: 4
Минимальная занятость диска: 40
ID компьютера: 2
Минимальная занятость диска: 60

Task 3

Жёсткий диск: 1
Компьютер: 1
Жёсткий диск: 1
Компьютер: 3
Жёсткий диск: 2
Компьютер: 1
Жёсткий диск: 3
Компьютер: 1
Жёсткий диск: 3
Компьютер: 3
Жёсткий диск: 3
Компьютер: 4
Жёсткий диск: 4
Компьютер: 2
Жёсткий диск: 5
Компьютер: 2
Жёсткий диск: 5
Компьютер: 4
Жёсткий диск: 6
Компьютер: 3

Жёсткий диск: 7
Компьютер: 3
Жёсткий диск: 7
Компьютер: 1
Жёсткий диск: 7
Компьютер: 4
Жёсткий диск: 8
Компьютер: 4
Жёсткий диск: 9
Компьютер: 4

