**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»**

# **Курс «Базовые компоненты интернет-технологий»**

## Рубежный контроль №1

Выполнил:

студент группы ИУ5-33Б

Кузнецов В. А.

Проверил:

Преподаватель кафедры ИУ-5

Гапанюк Ю. Е.

2022 г.

**Текст программы**

from operator import itemgetter  
from math import inf  
  
  
class HardDrive:  
 *"""Жёсткий диск"""* def \_\_init\_\_(self, id, model, size, computer\_id):  
 self.id = id  
 self.model = model  
 self.size = size  
 self.computer\_id = computer\_id  
  
  
class Computer:  
 *"""Компьютер"""* def \_\_init\_\_(self, id, processor):  
 self.id = id  
 self.processor = processor  
  
  
class HDComp:  
 *"""  
 'Жёсткий диск компьютера' для реализации  
 связи многие-ко-многим  
 """* def \_\_init\_\_(self, computer\_id, hard\_drive\_id):  
 self.computer\_id = computer\_id  
 self.hard\_drive\_id = hard\_drive\_id  
  
  
*# Компьютеры*computers = [  
 Computer(1, 'Intel Core i3-6100'),  
 Computer(2, 'Intel Core i3-7300'),  
 Computer(3, 'Intel Core i5-7400'),  
  
 Computer(11, 'Intel Core i3-7100'),  
 Computer(22, 'AMD Ryzen 3 3100'),  
 Computer(33, 'Intel Core i5-6500'),  
]  
  
*# Жёсткие диски*hard\_drives = [  
 HardDrive(1, 'Western Digital Blue WD10EZEX', 1024, 1),  
 HardDrive(2, 'Seagate Barracuda ST500LM030', 500, 2),  
 HardDrive(3, 'Western Digital Purple WD40PURX', 4096, 3),  
 HardDrive(4, 'Seagate SkyHawk ST2000VX008', 2048, 3),  
 HardDrive(5, 'Western Digital Blue WD5000LPCX', 500, 3),  
]  
  
hds\_comps = [  
 HDComp(1, 1),  
 HDComp(2, 2),  
 HDComp(3, 3),  
 HDComp(3, 4),  
 HDComp(3, 5),  
  
 HDComp(11, 1),  
 HDComp(22, 2),  
 HDComp(33, 3),  
 HDComp(33, 4),  
 HDComp(33, 5),  
]  
  
  
def print\_by\_line(data):  
 for line in data:  
 print(line)  
  
  
def task\_1(one\_to\_many):  
 *"""  
 «Компьютер» и «Жёсткий диск» связаны соотношением один-ко-многим.  
 Выведите список всех жёстких дисков фирмы «Western Digital»,  
 и названия процессоров компьютеров, в которые они установлены.  
 """* print('Задание А1')  
 print\_by\_line([(record[0], record[2]) for record in one\_to\_many if record[0].startswith('Western Digital')])  
  
  
def task\_2(one\_to\_many):  
 *"""  
 «Компьютер» и «Жёсткий диск» связаны соотношением один-ко-многим.  
 Выведите список процессоров компьютеров с минимальным объёмом диска в каждом компьютере,  
 отсортированный по минимальному объёму диска.  
 """* print('\nЗадание А2')  
 mins = {}  
 for model, size, processor in one\_to\_many:  
 mins[processor] = min(mins.get(processor, inf), size)  
 print\_by\_line(sorted(mins.items(), key=itemgetter(1)))  
  
  
def task\_3(many\_to\_many):  
 *"""  
 «Компьютер» и «Жёсткий диск» связаны соотношением многие-ко-многим.  
 Выведите список всех связанных жёстких дисков и компьютеров,  
 отсортированный по объёму жёстких дисков, сортировка по компьютерам произвольная.  
 """* print('\nЗадание А3')  
 print\_by\_line(sorted(many\_to\_many, key=itemgetter(1)))

def main():  
 *"""Основная функция"""  
  
 # Соединение данных один-ко-многим* one\_to\_many = [(hard\_drive.model, hard\_drive.size, computer.processor)  
 for hard\_drive in hard\_drives  
 for computer in computers  
 if hard\_drive.computer\_id == computer.id]  
  
 *# Соединение данных многие-ко-многим* many\_to\_many\_temp = [(computer.processor, hd\_c.computer\_id, hd\_c.hard\_drive\_id)  
 for computer in computers  
 for hd\_c in hds\_comps  
 if computer.id == hd\_c.computer\_id]  
  
 many\_to\_many = [(hard\_drive.model, hard\_drive.size, processor)  
 for processor, computer\_id, hard\_drive\_id in many\_to\_many\_temp  
 for hard\_drive in hard\_drives  
 if hard\_drive.id == hard\_drive\_id]  
  
 task\_1(one\_to\_many)  
 task\_2(one\_to\_many)  
 task\_3(many\_to\_many)  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 main()

**Результат выполнения**

