# МГТУ имени Баумана

Факультет «Информатика и Системы управления»

# Кафедра «Системы обработки информации и управления» Дисциплина «Разработка интернет-приложений»

«Основные конструкции языка Python»

# Рубежный контроль

Выполнил: Кашурин Максим, ИУ5-54Б

Проверил: Гапанюк Ю.Е.

# Текст программы:

import operator  
from operator import itemgetter  
  
  
class HardDisk:  
   
  
 def \_\_init\_\_(self, id, marka, price, pcc\_id):  
 self.id = id  
 self.marka = marka  
 self.price = price  
 self.pcc\_id = pcc\_id  
  
  
class PC:  
   
  
 def \_\_init\_\_(self, id, name):  
 self.id = id  
 self.name = name  
  
  
class DiskPC:  
   
  
 def \_\_init\_\_(self, pcc\_id,disk\_id ):  
 self.pcc\_id = pcc\_id  
 self.disk\_id = disk\_id  
  
  
   
pcs= [  
 PC(1, 'Apple'),  
 PC(2, 'Asus'),  
 PC(3, 'Lenovo'),  
  
 PC(11, 'Honor'),  
 PC(22, 'HP'),  
 PC(33, 'Huawei'),  
]  
  
   
disks = [  
 HardDisk(1, 'Apple inc', 55000, 1),  
 HardDisk(2, 'Atasi', 81000, 2),  
 HardDisk(3, 'Cogito Systems', 120000, 3),  
 HardDisk(4, 'Fuji', 80000, 3),  
 HardDisk(5, 'IBM', 40000, 3),  
]  
  
disks\_pcs = [  
 DiskPC(1, 1),  
 DiskPC(2, 2),  
 DiskPC(3, 3),  
 DiskPC(3, 4),  
 DiskPC(3, 5),  
  
 DiskPC(11, 1),  
 DiskPC(22, 2),  
 DiskPC(33, 3),  
 DiskPC(33, 4),  
 DiskPC(33, 5),  
]  
  
  
def main():  
 *"""Основная функция"""* # Соединение данных один-ко-многим  
 one\_to\_many = [(e.marka, e.price, d.name)  
 for d in pcs  
 for e in disks  
 if e.pcc\_id == d.id]  
  
 # Соединение данных многие-ко-многим  
 many\_to\_many\_temp = [(d.name, ed.pcc\_id, ed.disk\_id)  
 for d in pcs  
 for ed in disks\_pcs  
 if d.id == ed.pcc\_id]  
  
 many\_to\_many = [(e.marka, dep\_name)  
 for dep\_name, dep\_id, emp\_id in many\_to\_many\_temp  
 for e in disks if e.id == emp\_id]  
  
 print('Задание А1')  
 for d in pcs:  
 if (d.name[0]!='А'):  
 print(d.name, ':')  
 for i in disks:  
 if i.pcc\_id==d.id:  
 print(i.marka)  
  
 print('\nЗадание А2')  
 res\_12\_unsorted = []  
 for d in pcs:  
 d\_emps = list(filter(lambda i: i[2] == d.name, one\_to\_many))  
 if len(d\_emps) > 0:  
 d\_sals = [sal for \_, sal, \_ in d\_emps]  
   
 d\_sals\_max = max(d\_sals)  
 res\_12\_unsorted.append((d.name, d\_sals\_max))  
   
 res\_12 = sorted(res\_12\_unsorted, key=itemgetter(1), reverse=True)  
 print(res\_12)  
  
  
  
 print('\nЗадание А3')  
 res\_11 = sorted(many\_to\_many, key=itemgetter(1))  
 print(res\_11)  
  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 main()

# Результат выполнения:

