ИУ5-32Б,

Колесников Никита

**Рубежный контроль 1**

Номер варианта:



Вариант Б:

1. «Операционная система» и «Компьютер» связаны соотношением один ко-многим. Выведите список всех связанных компьютеров и операционных систем, отсортированный по Компьютерам, сортировка по операционных систем произвольная.
2. «Операционная система» и «Компьютер» связаны соотношением один ко-многим. Выведите список компьютеров с количеством операционных систем в каждом компьютере, отсортированный по количеству операционных систем.
3. «Операционная система» и «Компьютер» связаны соотношением многие ко-многим. Выведите список все компьютеры, у которых название заканчивается на «s», и названия их операционных систем.

**Текст программы:**

# используется для сортировки  
from operator import itemgetter  
  
class OS:  
 *"""Операционная система"""* def \_\_init\_\_(self, id, name, version, comp\_id):  
 self.id = id  
 self.name = name  
 self.version = version  
 self.comp\_id = comp\_id  
  
class Comp:  
 *"""Компьютер"""* def \_\_init\_\_(self, id, name):  
 self.id = id  
 self.name = name  
  
class OSComp:  
 *"""  
 'Операционная система компьютера' для реализации  
 связи многие-ко-многим  
 """* def \_\_init\_\_(self, os\_id, comp\_id):  
 self.os\_id = os\_id  
 self.comp\_id = comp\_id  
  
# Компьютеры  
comps = [  
 Comp(1, 'Apple MacBook Air'),  
 Comp(2, 'Dell XPS 13'),  
 Comp(3, 'Acer Swift 3'),  
]  
# Операционные системы  
OSs = [  
 OS(1, 'Windows', 11, 1),  
 OS(2, 'macOS', 14, 2),  
 OS(3, 'Android', 14, 3),  
 OS(4, 'iOS', 17, 3),  
 OS(5, 'Linux', 23.10, 3),  
]  
comps\_OSs = [  
 OSComp(1,1),  
 OSComp(2,2),  
 OSComp(5,3),  
 OSComp(3,1),  
 OSComp(4,3)  
]  
  
def main():# Соединение данных один-ко-многим  
 one\_to\_many = [(os.name, os.version, c.name)  
 for c in comps  
 for os in OSs  
 if os.comp\_id == c.id]  
  
 # Соединение данных многие-ко-многим  
 many\_to\_many\_temp = [(c.name, cos.comp\_id, cos.os\_id)  
 for c in comps  
 for cos in comps\_OSs  
 if c.id == cos.comp\_id]  
  
 many\_to\_many = [(os.name, os.version, comp\_name)  
 for comp\_name, comp\_id, os\_id in many\_to\_many\_temp  
 for os in OSs if os.id == os\_id]  
  
 print('Задание Б1')  
 res\_1 = sorted(one\_to\_many, key=itemgetter(2))  
 print(res\_1)  
  
 print('\nЗадание Б2')  
 res\_2 = []  
 temp\_dict=dict()  
 for i in one\_to\_many:  
 if i[2] in temp\_dict:  
 temp\_dict[i[2]]+=1  
 else:  
 temp\_dict[i[2]]=1  
 for i in temp\_dict.keys():  
 res\_2.append((i,temp\_dict[i]))  
 res\_2.sort(key=itemgetter(1), reverse=True)  
 print(res\_2)  
  
 print('\nЗадание Б3')  
 res\_3=[]  
 for i in many\_to\_many:  
 if i[0][-1]=='S':  
 res\_3.append((i[0], i[2]))  
 print(res\_3)  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 main()

**Анализ результатов:**

