**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 предмету

«Базовые компоненты интернет-технологий»

Выполнил:

студент группы ИУ5-31Б

Михалёв Ярослав

Проверил:

Преподаватель кафедры ИУ-5

Гапанюк Юрий

2022 г.

**Вариант А. Предметная область №12**

1. «Язык программирования» и «Средство разработки» связаны соотношением один-ко-многим. Выведите список всех связанных языков и IDE, отсортированных по IDE, сортировка по языкам произвольная.
2. «Язык программирования» и «Средство разработки» связаны соотношением один-ко-многим. Выведите список IDE, которые поддерживают больше всего языков. Вывод совершить в порядке возрастания
3. «Язык программирования» и «Средство разработки» связаны соотношением многие-ко-многим. Выведите список всех IDE, название которых начинается с 'V' и список поддерживаемых ими языков.

**Код программы**

**Main.py**

*from* operator *import* itemgetter  
  
  
*class* Lang:  
 """Язык программирования"""  
 *def \_\_init\_\_*(self, *id*, *name*, *version*):  
 self.id = *id* self.name = *name* self.version = *version  
  
class* IDE:  
 """Среда разработки"""  
 *def \_\_init\_\_*(self, *id*, *name*, *version*, *usersCount*, *\*args*):  
 self.id = *id* self.name = *name* self.version = *version* self.usersCount = *usersCount* self.langs = *args  
  
class* IdeLanguage:  
 """   
 'Установленные языки программирования' для реализации   
 связи многие-ко-многим  
 """  
 *def \_\_init\_\_*(self, *lang\_id*, *ide\_id*):  
 self.lang\_id = *lang\_id* self.ide\_id = *ide\_id*

# Языки программирования  
LANGUAGES = [  
 Lang(1, "Python", "3.1.0"),  
 Lang(2, "C#", "10.0"),  
 Lang(3, "JavaScript", "1.8.5"),  
 Lang(4, "C++", "20.0"),  
 Lang(5, "Rust", "1.56.0"),  
 Lang(6, "Java", "11.0"),  
 Lang(7, "Go", "11.0"),  
 Lang(8, "Swift", "11.0")  
]  
  
# Среды разработки  
IDEs = [  
 IDE(1, "PyCharm", "3.5.3", 543354, 1, 4, 6, 7),  
 IDE(2, "Visual Studio", "1.0.2", 8132032, 2, 5, 4, 8),  
 IDE(3, "Sublime Text", "3.8.4",3855844, 1, 3, 6, 7, 8),  
 IDE(4, "Notepad", "2.9.4", 7256184, 2, 4, 6),  
 IDE(5, "WebStorm", "0.6.3", 2616095, 3, 5),  
 IDE(6, "Visual Studio Code", "2.0.5", 696233, 4, 6, 7, 8)  
]

IDE\_LANGUAGES = [  
 IdeLanguage(0, 4),  
 IdeLanguage(1, 6),  
 IdeLanguage(2, 0),  
 IdeLanguage(3, 3),  
 IdeLanguage(4, 5),  
 IdeLanguage(6, 2)  
]  
  
*def* task\_1(*one\_to\_many*):  
 """ ЗАДАНИЕ №1.  
 Вывести список всех языков и IDE, отсортированных по количеству пользователей.  
 """  
 *return* sorted(*one\_to\_many*, key=itemgetter(1))  
  
  
*def* task\_2(*one\_to\_many*):  
 """ ЗАДАНИЕ №2.  
 Вывести список IDE, которое поддерживает больше всего языков. Вывод совершить в порядке возрастания.  
 """  
 *return* sorted([(name, len(langs)) *for* name, usersCount, langs *in one\_to\_many*],  
 key=itemgetter(1),  
 reverse=*False*)  
  
  
*def* task\_3(*many\_to\_many*):  
 """ ЗАДАНИЕ №3.  
 Вывести список всех IDE, название которых начинается с 'V' и список поддерживаемых ими языков  
 """  
 *return* [(name, lst) *for* name, count, lst *in* list(filter(*lambda el*: el[0][0] == 'V', *many\_to\_many*))]

*def* main():  
 """Основная функция"""  
  
 # Соединение данных один-ко-многим   
 one\_to\_many = [  
 (ide.name, ide.usersCount,  
 [lang.name *for* lang\_id *in* ide.langs *for* lang *in* LANGUAGES *if* lang.id == lang\_id])  
 *for* ide *in* IDEs  
 ]  
  
 # Соединение данных многие-ко-многим  
 many\_to\_many\_temp = [  
 (language.name, language.id, ide\_language.ide\_id)  
 *for* language *in* LANGUAGES  
 *for* ide\_language *in* IDE\_LANGUAGES  
 *if* language.id == ide\_language.lang\_id  
 ]  
  
 many\_to\_many = [  
 (ide\_temp.name, ide\_temp.usersCount,  
 [lang.name *for* el *in* ide\_temp.langs *for* lang *in* LANGUAGES *if* lang.id == el])  
 *for* name\_language, languageID, ideID *in* many\_to\_many\_temp  
 *for* ide\_temp *in* IDEs  
 *if* ide\_temp.id == ideID  
 ]  
  
 # print(\*one\_to\_many, sep='\n', end=f'\n {"=" \* 30}\n')  
 # print(\*many\_to\_many, sep='\n', end=f'\n {"=" \* 30}\n')  
  
 print(f'{"-" \* 10} Задание №1. {"-" \* 10}')  
 print(\*task\_1(one\_to\_many), sep='\n', end='\n\n')  
  
 print(f'{"-" \* 10} Задание №2. {"-" \* 10}')  
 print(\*task\_2(one\_to\_many), sep='\n', end='\n\n')  
  
 print(f'{"-" \* 10} Задание №3. {"-" \* 10}')  
 print(\*task\_3(many\_to\_many), sep='\n', end='\n\n')  
  
*if* \_\_name\_\_ == '\_\_main\_\_':  
 main()

**Результат выполнения**

