|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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»

ДИСЦИПЛИНА:

«БКИТ»

**Рубежная контроль № 1**

Студент Ткаченко В.С. ИУ5Ц-54Б **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

(И.О. Фамилия) (Группа) (Подпись, дата)

Преподаватель Гапанюк Ю.Е.  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

(И.О. Фамилия) (Подпись, дата)

Москва, 2021г.

**Задание дано:**

**Вариант Г.**

1. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список всех отделов, у которых название начинается с буквы «А», и список работающих в них сотрудников.
2. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список отделов с максимальной зарплатой сотрудников в каждом отделе, отсортированный по максимальной зарплате.
3. «Отдел» и «Сотрудник» связаны соотношением многие-ко-многим. Выведите список всех связанных сотрудников и отделов, отсортированный по отделам, сортировка по сотрудникам произвольная.

**Вариант:**

|  |  |  |
| --- | --- | --- |
| 22 | Библиотека | Язык программирования |

**Код программы:**

|  |
| --- |
| from operator import itemgetter  class Language:  def \_\_init\_\_(self, id\_lan, l):  self.id\_lan = id\_lan  self.l = l  class Library:  def \_\_init\_\_(self, id\_det, name, soft\_cost, id\_lan):  self.id\_lan = id\_lan  self.id\_det = id\_det  self.name = name  self.soft\_cost = soft\_cost  class LibLan:  def \_\_init\_\_(self, id\_det, id\_lan):  self.id\_det = id\_det  self.id\_lan = id\_lan  libs = [  Library(1, 'Emoji', 500, 1),  Library(2, 'Pandas', 700, 1),  Library(3, 'Console Paint', 1000, 2),  Library(11, 'GraphABC', 1500, 3),  Library(22, 'Math', 1200, 4),  Library(33, 'Guava', 2000, 5), ] langs = [  Language(1, 'Python'),  Language(2, 'C#'),  Language(3, 'Pascal'),  Language(4, 'AC++'),  Language(5, 'Java'), ] cross = [  LibLan(1, 1),  LibLan(2, 2),  LibLan(3, 3),  LibLan(3, 4),  LibLan(3, 5),  LibLan(11, 1),  LibLan(22, 2),  LibLan(33, 3),  LibLan(33, 4),  LibLan(33, 5), ]  def main():  *"""Основная функция"""* one\_to\_many = [(e.name, e.soft\_cost, d.l) for d in langs for e in libs  if e.id\_lan == d.id\_lan]   many\_to\_many\_temp = [(d.l, ed.id\_lan, ed.id\_det) for d in langs for ed in cross  if d.id\_lan == ed.id\_lan]   many\_to\_many = [(e.name, e.soft\_cost, lang\_name) for lang\_name, id\_lan, id\_det in many\_to\_many\_temp for e in libs if  e.id\_det == id\_det]  print('Задание №1') res1 = {} for d in langs:  if d.l.lower().startswith('a'):  d\_emps = list(filter(lambda i: i[2] == d.l, one\_to\_many))  d\_emps\_name = [x for x, \_, \_ in d\_emps] res1[d.l] = d\_emps\_name print(res1)  print('\nЗадание №2') res\_12\_unsorted = [] for d in langs:  d\_emps = list(filter(lambda i: i[2] == d.l, many\_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.l, d\_sals\_max))   res\_12 = sorted(res\_12\_unsorted, key=itemgetter(1), reverse=True) print(res\_12)  print('Задание №3') res\_11 = sorted(many\_to\_many, key=itemgetter(2)) print(res\_11) if \_\_name\_\_ == '\_\_main\_\_':  main() |

**Результат программы:**

|  |
| --- |
| Задание №1  {'AC++': ['Math']}  Задание №2  [('Pascal', 2000), ('AC++', 2000), ('Java', 2000), ('Python', 1500), ('C#', 1200)]  Задание №3  [('Console Paint', 1000, 'AC++'), ('Guava', 2000, 'AC++'), ('Pandas', 700, 'C#'), ('Math', 1200, 'C#'), ('Console Paint', 1000, 'Java'), ('Guava', 2000, 'Java'), ('Console Paint', 1000, 'Pascal'), ('Guava', 2000, 'Pascal'), ('Emoji', 500, 'Python'), ('GraphABC', 1500, 'Python')] |