**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»

Отчет по РК №2

Вариант запросов: B

Вариант предметной области: 12

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы ИУ5-32Б |  | преподаватель каф. ИУ5 |
| Кузнецов Егор |  | Гапанюк Ю. Е. |
|  |  |  |

Москва, 2023 г.

**Решение варианта В для предметной области 12**

1. «Язык программирования» и «Средство разработки» связаны

соотношением один-ко-многим. Выведите список всех языков

программирования, у которых название начинается на «P».

2. «Язык программирования» и «Средство разработки» связаны

соотношением один-ко-многим. Выведите список аудиторий со средней

популярностью поддерживаемых языков для каждого инструмента

разработки в порядке возрастания.

3. «Язык программирования» и «Средство разработки» связаны

соотношением многие-ко-многим. Выведите связи между языками и

средами разработки, отсортированные сначала по названию языка, а

потом по популярности.

**Текст программы**

Main.py

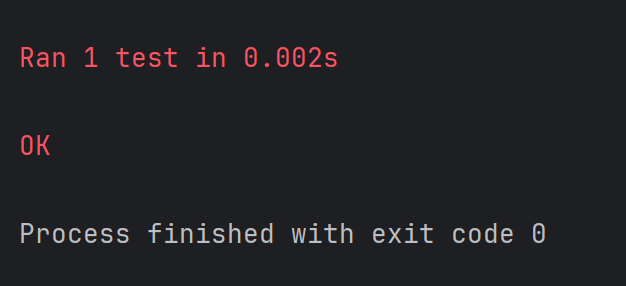
from operator import itemgetter  
  
class ProgrammingLanguage:  
 def \_\_init\_\_(self, id, name, instr\_id, popularity):  
 self.id = id  
 self.name = name  
 self.instr\_id = instr\_id  
 self.popularity = popularity  
  
class DevelopmentTool:  
 def \_\_init\_\_(self, id, name):  
 self.id = id  
 self.name = name  
  
class Language\_Tool:  
 def \_\_init\_\_(self, instr\_id, lang\_id):  
 self.instr\_id = instr\_id  
 self.lang\_id = lang\_id  
  
# Данные о языках программирования  
languages = [  
 ProgrammingLanguage(1, 'C++', 3, 2.8),  
 ProgrammingLanguage(2, 'Python', 2, 13.4),  
 ProgrammingLanguage(3, 'Java', 1, 14),  
 ProgrammingLanguage(4, 'C', 4, 0.7),  
 ProgrammingLanguage(5, 'Go', 4, 1.9),  
 ProgrammingLanguage(6, 'PHP', 5, 7.1),  
 ProgrammingLanguage(7, 'JavaScript', 4, 19.1)  
]  
  
# Данные о средствах разработки  
development\_tools = [  
 DevelopmentTool(1, 'IntelliJ IDEA'),  
 DevelopmentTool(2, 'PyCharm'),  
 DevelopmentTool(3, 'CLion' ),  
 DevelopmentTool(4, 'VSCode'),  
 DevelopmentTool(5, 'PhP\_sreda')  
]  
  
# Связи между языками и средствами разработки  
language\_tools = [  
 Language\_Tool(1, 3),  
 Language\_Tool(2, 2),  
 Language\_Tool(3, 1),  
 Language\_Tool(4, 3),  
 Language\_Tool(4, 4),  
 Language\_Tool(4, 5),  
 Language\_Tool(5, 6),  
 Language\_Tool(4, 7)  
]  
  
def main():  
 one\_to\_many = [(l.name, l.popularity, i.name)  
 for l in languages  
 for i in development\_tools  
 if l.instr\_id == i.id]  
 many\_to\_many\_temp = [(i.name, li.instr\_id, li.lang\_id)  
 for i in development\_tools  
 for li in language\_tools  
 if i.id == li.instr\_id  
 ]  
 many\_to\_many = [(l.name, l.popularity, instr\_name)  
 for instr\_name, instr\_id, lang\_id in many\_to\_many\_temp  
 for l in languages if l.id == lang\_id]  
 print('Task 1')  
 res\_1 = []  
 for i in one\_to\_many:  
 if i[0][0] == 'P':  
 res\_1.append(i)  
 print(res\_1)  
  
 print('Task 2')  
 res\_12\_unsorted = []  
 for instr in development\_tools:  
 devel\_langs = list(filter(lambda i: i[2] == instr.name, one\_to\_many))  
 if len(devel\_langs) > 0:  
 popularities = [popularity for \_, popularity, \_ in devel\_langs]  
 avg\_popularity = sum(popularities) / len(popularities)  
 res\_12\_unsorted.append((instr.name, avg\_popularity))  
 res\_12 = sorted(res\_12\_unsorted, key=itemgetter(1))  
 for tool, avg\_popularity in res\_12:  
 print(f"Среда разработки: {tool}, Средняя популярность поддерживаемых языков: {avg\_popularity:.2f}%")  
  
 print('Task 3')  
 print(sorted(many\_to\_many, key=lambda item: (item[0], item[1])))  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 main()

tests.py

import unittest  
from main import \*  
  
  
class Rk2\_test(unittest.TestCase):  
 languages = [  
 ProgrammingLanguage(1, 'C++', 3, 2.8),  
 ProgrammingLanguage(2, 'Python', 2, 13.4),  
 ProgrammingLanguage(3, 'Java', 1, 14),  
 ProgrammingLanguage(4, 'C', 4, 0.7),  
 ProgrammingLanguage(5, 'Go', 4, 1.9),  
 ProgrammingLanguage(6, 'PHP', 5, 7.1),  
 ProgrammingLanguage(7, 'JavaScript', 4, 19.1)  
 ]  
 development\_tools = [  
 DevelopmentTool(1, 'IntelliJ IDEA'),  
 DevelopmentTool(2, 'PyCharm'),  
 DevelopmentTool(3, 'CLion'),  
 DevelopmentTool(4, 'VSCode'),  
 DevelopmentTool(5, 'PhP\_sreda')  
 ]  
 language\_tools = [  
 Language\_Tool(1, 3),  
 Language\_Tool(2, 2),  
 Language\_Tool(3, 1),  
 Language\_Tool(4, 3),  
 Language\_Tool(4, 4),  
 Language\_Tool(4, 5),  
 Language\_Tool(5, 6),  
 Language\_Tool(4, 7)  
 ]  
 def test\_1(self):  
 one\_to\_many = [(l.name, l.popularity, i.name)  
 for l in self.languages  
 for i in self.development\_tools  
 if l.instr\_id == i.id]  
 res\_11 = []  
 for i in one\_to\_many:  
 if i[0][0] == 'P':  
 res\_11.append(i)  
 self.assertEqual(res\_11, [('Java', 14, 'IntelliJ')])  
  
 def test\_2(self):  
 one\_to\_many = [(l.name, l.popularity, i.name)  
 for l in self.languages  
 for i in self.development\_tools  
 if l.instr\_id == i.id]  
 res\_12\_unsorted = []  
 for instr in development\_tools:  
 devel\_langs = list(filter(lambda i: i[2] == instr.name, one\_to\_many))  
 if len(devel\_langs) > 0:  
 popularities = [popularity for \_, popularity, \_ in devel\_langs]  
 avg\_popularity = sum(popularities) / len(popularities)  
 res\_12\_unsorted.append((instr.name, avg\_popularity))  
 res\_12 = sorted(res\_12\_unsorted, key=itemgetter(1))  
 self.assertEqual(res\_12, 'Среда разработки: CLion, Средняя популярность поддерживаемых языков: 2.80%Среда разработки: PhP\_sreda, Средняя популярность поддерживаемых языков: 7.10%Среда разработки: VSCode, Средняя популярность поддерживаемых языков: 7.23%Среда разработки: PyCharm, Средняя популярность поддерживаемых языков: 13.40%Среда разработки: IntelliJ IDEA, Средняя популярность поддерживаемых языков: 14.00%')  
  
 def test\_3(self):  
 many\_to\_many\_temp = [(i.name, li.instr\_id, li.lang\_id)  
 for i in self.development\_tools  
 for li in self.language\_tools  
 if i.id == li.instr\_id  
 ]  
 many\_to\_many = [(l.name, l.popularity, instr\_name)  
 for instr\_name, instr\_id, lang\_id in many\_to\_many\_temp  
 for l in languages if l.id == lang\_id  
 ]  
 res = sorted(many\_to\_many, key=lambda item: (item[0], item[1]))  
 self.assertEqual(res, [('C', 0.7, 'VSCode'), ('C++', 2.8, 'CLion'), ('Go', 1.9, 'VSCode'), ('Java', 14, 'IntelliJ IDEA'), ('Java', 14,  
'VSCode'), ('JavaScript', 19.1, 'VSCode'), ('PHP', 7.1, 'PhP\_sreda'), ('Python', 13.4, 'PyCharm')])  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 unittest.main()

Результаты работы.

Все тесты пройдены успешно

****