***Текст программы***

# используется для сортировки

from operator import itemgetter

class Lang:

    """Язык программирования"""

    def \_\_init\_\_(self, id, name, rating, tool\_id):

        self.id = id

        self.name = name

        self.rating = rating

        self.tool\_id = tool\_id

class DevTool:

    """Средство разработки"""

    def \_\_init\_\_(self, id, name):

        self.id = id

        self.name = name

class LangDevTool:

    """

    'Средства разработки языков программирования' для реализации

    связи многие-ко-многим

    """

    def \_\_init\_\_(self, lang\_id, tool\_id):

        self.lang\_id = lang\_id

        self.tool\_id = tool\_id

# Средства разработки

devTools = [

    DevTool(1, 'Android Studio'),

    DevTool(2, 'IntelliJ IDEA'),

    DevTool(3, 'Visual Studio Code 2020'),

    DevTool(4, 'Visual Studio 2019'),

    DevTool(13, 'CLion'),

    DevTool(20, 'PyCharm'),

]

# Языки программирования

langs = [

    Lang(1, 'Java', 95.3, 1),

    Lang(2, 'JavaScript', 79.5, 3),

    Lang(5, 'C++', 87, 4),

    Lang(11, 'C#', 48.1, 4),

    Lang(19, 'Python', 100, 20),

    Lang(31, 'F#', 34.7, 4),

]

langs\_devTools = [

    LangDevTool(1,1),

    LangDevTool(1,2),

    LangDevTool(2,3),

    LangDevTool(5,4),

    LangDevTool(5,13),

    LangDevTool(11,4),

    LangDevTool(19,3),

    LangDevTool(19,4),

    LangDevTool(19,20),

    LangDevTool(31,4),

]

def main():

    """Основная функция"""

    # Соединение данных один-ко-многим

    one\_to\_many = [(l.name, l.rating, t.name)

        for l in langs

        for t in devTools

        if l.tool\_id==t.id]

    # Соединение данных многие-ко-многим

    many\_to\_many\_temp = [(t.name, ldt.lang\_id, ldt.tool\_id)

        for t in devTools

        for ldt in langs\_devTools

        if t.id==ldt.tool\_id]

    many\_to\_many = [(l.name, l.rating, tool\_name)

        for tool\_name, lang\_id, tool\_id in many\_to\_many\_temp

        for l in langs

        if l.id==lang\_id]

    print('Задание D1')

    res1 = []

    for i in one\_to\_many:

        if i[0].endswith('#') ==True:

            res1.append(i[0:3:2])

    print(res1)

    print('\nЗадание D2')

    res\_12\_unsorted = []

    # Перебираем все средства разработки(инструменты)

    for t in devTools:

        # Список языков, поддерживаемых инструментом разработки

        t\_langs = list(filter(lambda i: i[2]==t.name, one\_to\_many))

        if len(t\_langs) > 0:

            # Находим средний рейтинг языков для каждого инструмента

            t\_rating = [rating for \_,rating,\_ in t\_langs]

            t\_rating\_sum = sum(t\_rating)

            t\_rating\_count = len(t\_rating)

            t\_rating\_average = t\_rating\_sum / t\_rating\_count

            res\_12\_unsorted.append((t.name, t\_rating\_average))

    # Сортировка по среднему рейтингу

    res\_12 = sorted(res\_12\_unsorted, key=itemgetter(1), reverse=True)

    print(res\_12)

    print('\nЗадание Д3')

    res3 = {}

    # Перебираем все средства разработки(инструменты)

    for t in devTools:

        #Выбираем инструменты, название которых начинается на букву V

        if t.name[0] == "V":

            t\_langs = list(filter(lambda i: i[2] == t.name, many\_to\_many))

            # Только название языка

            o\_langs\_names = [x for x, \_, \_ in t\_langs]

            # Добавляем результат в словарь

            # ключ - инструмент, значение - список языков

            res3[t.name] = o\_langs\_names

    print(res3)

if \_\_name\_\_ == '\_\_main\_\_':

    main()

***Результаты выполнения***

Задание D1

[('C#', 'Visual Studio 2019'), ('F#', 'Visual Studio 2019')]

Задание D2

[('PyCharm', 100.0), ('Android Studio', 95.3), ('Visual Studio Code 2020', 79.5), ('Visual Studio 2019', 56.6)]

Задание Д3

{'Visual Studio Code 2020': ['JavaScript', 'Python'], 'Visual Studio 2019': ['C++', 'C#', 'Python', 'F#']}

