Предметная область Е, вариант 21. Классы: Язык программирования, Оператор.

Задания:

1. «Язык программирования» и «Оператор» связаны соотношением один-ко-многим. Выведите список всех языков, которые начинаются с буквы «P», и список его операторов.

2. «Язык программирования» и «Оператор» связаны соотношением один-ко-многим. Выведите список языков со средним количеством букв в названии операторов, отсортированный по среднему количеству букв. Среднее количество букв должна быть округлено до 2 знаков после запятой.

3. «Язык программирования» и «Оператор» связаны соотношением многие-ко-многим. Выведите список всех операторов, у которых название начинается с буквы «w», и названия их языков.

# используется для сортировки

from operator import itemgetter

class Operatop:

"""Оператор"""

def \_\_init\_\_(self, id, name, typ, num, lan\_id):

self.id = id

self.name = name

self.typ = typ

self.num = num

self.lan\_id = lan\_id

class Language:

"""Язык программирования"""

def \_\_init\_\_(self, id, name):

self.id = id

self.name = name

class OpLan:

"""

'Оператор языка программирования' для реализации

связи многие-ко-многим

"""

def \_\_init\_\_(self, lan\_id, op\_id):

self.lan\_id = lan\_id

self.op\_id = op\_id

# Языки

Languages = [

Language(1, 'Python'),

Language(2, 'C++'),

Language(3, 'Pascal'),

Language(4, 'Java'),

Language(5, 'C'),

Language(6, 'Delphi'),

]

# Операторы

Operatops = [

Operatop(1, 'switch/case', 'Условие', 10, 1),

Operatop(2, '(type)', 'Приведение типа', 4, 2),

Operatop(3, 'Goto', 'Безусловный переход', 4, 3),

Operatop(4, 'while', 'Цикл', 5, 1),

Operatop(5, 'if', 'Условие', 2, 1),

]

Operatops\_Languages = [

OpLan(1,1),

OpLan(4,1),

OpLan(1,4),

OpLan(2,4),

OpLan(3,4),

OpLan(4,4),

OpLan(5,4),

OpLan(6,4),

OpLan(1,5),

OpLan(2,5),

OpLan(3,5),

OpLan(4,5),

OpLan(5,5),

OpLan(6,5),

OpLan(2,2),

OpLan(5,2),

OpLan(3,3),

OpLan(6,3),

]

def main():

"""Основная функция"""

# Соединение данных один-ко-многим

one\_to\_many = [(e.name, e.num, e.typ, d.name)

for d in Languages

for e in Operatops

if e.lan\_id==d.id]

# Соединение данных многие-ко-многим

many\_to\_many\_temp = [(d.name, ed.lan\_id, ed.op\_id)

for d in Languages

for ed in Operatops\_Languages

if d.id==ed.lan\_id]

many\_to\_many = [(e.name, e.num, lan\_name)

for lan\_name, lan\_id, op\_id in many\_to\_many\_temp

for e in Operatops if e.id==op\_id]

print('Задание E1')

res\_1 = list(filter(lambda x: 'P' in x[3], one\_to\_many))

were = ''

for elem in res\_1:

if(elem[2] != were):

were = elem[3]

print(were, ":", sep='')

if(elem[3] == were):

print(' ', elem[0], elem[1], elem[2])

print('\nЗадание E2')

avg\_len = dict()

for link in one\_to\_many:

if (link[3] in avg\_len):

avg\_len[link[3]].append(link[1])

else:

avg\_len[link[3]] = [link[1]]

for key, value in avg\_len.items():

print(key, round(sum(value) / len(value), 2))

print('\nЗадание E3')

res\_3 = list(filter(lambda x: x[0][0] == 'w', many\_to\_many))

for i in range(len(res\_3)):

print(str(i + 1) + '.', res\_3[i][0], res\_3[i][2])

if \_\_name\_\_ == '\_\_main\_\_':

main()

Результаты:

