

Рубежного контроль №2 по курсу БКИТ. Вариант 6Д

Классы: «Дом» - «Улица»

Задание:

1. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим.
Выведите список всех сотрудников, у которых фамилия заканчивается на «ов», и названия их отделов.
2. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим.
Выведите список отделов со средней зарплатой сотрудников в каждом отделе, отсортированный по средней зарплате (отдельной функции вычисления среднего значения в Python нет, нужно использовать комбинацию функций вычисления суммы и количества значений).
3. «Отдел» и «Сотрудник» связаны соотношением многие-ко-многим.
Выведите список всех отделов, у которых название начинается с буквы «А», и список работающих в них сотрудников.

Код программы.

Изменённый код первого рк для модульного тестирования.

```
# используется для сортировки
from operator import itemgetter

class House:
    """Дом"""

    def __init__(self, id, number, entrances, street_id):
        self.id = id
        self.number = number
        self.entrances = entrances
        self.street_id = street_id

class Street:
    """Улица"""

    def __init__(self, id, name):
        self.id = id
        self.name = name

class HouseStreet:
    """
    'Дома улицы' для реализации
    связи многие-ко-многим
    """

    def __init__(self, street_id, house_id):
        self.street_id = street_id
```

```

        self.house_id = house_id

# Улицы
streets = [
    Street(1, 'Раздольная'),
    Street(2, 'Металлургов'),
    Street(3, 'Пушкинская'),
    Street(4, 'Измайловская'),
    Street(5, 'Бауманская'),
    Street(6, 'Аськина'),
    Street(7, 'Восточная'),
    Street(8, 'Алексеевская'),
]

# Дома
houses = [
    House(1, 12, 7, 1),
    House(2, 44, 3, 2),
    House(3, 15, 4, 3),
    House(4, 11, 3, 4),
    House(5, 12, 9, 5),
    House(6, 107, 4, 6),
    House(7, 12, 5, 6),
    House(8, 49, 4, 7),
    House(9, 36, 1, 7),
    House(10, 29, 4, 8),
]

houses_streets = [
    HouseStreet(1, 1),
    HouseStreet(2, 2),
    HouseStreet(3, 3),
    HouseStreet(4, 4),
    HouseStreet(5, 5),
    HouseStreet(6, 6),
    HouseStreet(6, 7),
    HouseStreet(7, 8),
    HouseStreet(7, 9),
    HouseStreet(8, 10),
]

# Соединение данных один-ко-многим
one_to_many = [(h.number, h.entrances, s.name)
                for h in houses
                for s in streets
                if h.street_id == s.id]

# Соединение данных многие-ко-многим
many_to_many_temp = [(s.name, hs.street_id, hs.house_id)
                      for s in streets
                      for hs in houses_streets
                      if s.id == hs.street_id]

many_to_many = [(h.number, h.entrances, street_name)
                 for street_name, street_id, house_id in many_to_many_temp
                 for h in houses if h.id == house_id]

def task1():
    return [el for el in one_to_many if el[0] == 12]

```

```

def task2():
    summa = 0
    count = 0
    avg = []
    for s in streets:
        for h in houses:
            if s.id == h.id:
                summa += h.entrances
                count += 1
        avg.append((s.name, round(summa / count, 2)))
    return sorted(avg, key=itemgetter(1))

def task3():
    names = (d.name for d in streets if d.name[:1] == 'А')
    return [(name, [s[0] for s in many_to_many if s[2] == name]) for name in
names]

```

Модульное тестирование.

```

import unittest
from main import *

task_1_result = [
    (12, 7, 'Раздольная'), (12, 9, 'Бауманская'), (12, 5, 'Аськина')
]

task_2_result = [
    ('Измайловская', 4.25),
    ('Пушкинская', 4.67),
    ('Алексеевская', 4.88),
    ('Металлургов', 5.0),
    ('Аськина', 5.0),
    ('Восточная', 5.0),
    ('Бауманская', 5.2),
    ('Раздольная', 7.0)
]

task_3_result = [
    ('Аськина', [107, 12]), ('Алексеевская', [29])
]

class RK_test(unittest.TestCase):
    def test_task_1(self):
        self.assertEqual(task_1_result, task1())

    def test_task_2(self):
        self.assertEqual(task_2_result, task2())

    def test_task_3(self):
        self.assertEqual(task_3_result, task3())

```

Результаты тестирования

Ran 3 tests in 0.003s

OK

Process finished with exit code 0