Московский государственный технический университет имени Н. Э. Баумана

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Факультет Радиотехнический

Кафедра ИУ5 Системы обработки информации и управления

**Отчет по рубежному контролю № 1**

**Базовые компоненты интернет-технологий**

**Изучение объектно-ориентированных возможностей языка Python**

**Вариант 12**

|  |  |  |
| --- | --- | --- |
| Исполнитель |  |  |
| студент группы РТ5-31Б | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Кабанец В. М. |
|  |  | “\_\_\_” \_\_\_\_\_\_\_\_\_\_ 2022 г. |
|  |  |  |
| Проверил |  |  |
| Доцент кафедры ИУ5 | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Гапанюк Ю.Е. |
|  |  | “\_\_\_” \_\_\_\_\_\_\_\_\_\_ 2022 г. |

Москва - 2022

**Предметная область 12**

**Вариант E**

Язык программирования (Language) и среда разработки (IDE) соединены соотношением один-ко-многим. Вывести все среды разработки, содержащие "Studio" и соответствующие им языки программирования, согласно language\_ide.

Language and IDE соединены соотношением один-ко-многим. Выведите список IDE со средней сложностью Language в каждом IDE, отсортированный по средней сложности. Средняя сложность должна быть округлена до 2 знака после запятой.

«IDE» и «Language» связаны соотношением многие-ко-многим. Выведите список всех language, у которых название начинается с буквы «S», и названия их IDE

**Листинг программы**

from operator import itemgetter

# Язык программирования

# Средство разработки

*class* Language:

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

*self*.name = *name*

*self*.id = *id*

*self*.diificulity = *diificulity*

*class* IDE:

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

*self*.name = *name*

*self*.id = *id*

*class* LanguageIDE:

*def* \_\_init\_\_(*self*, *language\_id*, *ide\_id*):

*self*.language\_id = *language\_id*

*self*.ide\_id = *ide\_id*

languages = [

    Language('Python', 1, 1),

    Language('C++', 2, 2),

    Language('C#', 3, 3),

    Language('Java', 4, 4),

    Language('JavaScript', 5, 5),

    Language('PHP', 6, 6),

    Language('Ruby', 7, 7),

    Language('Swift', 8, 8),

    Language('Go', 9, 9),

    Language('Rust', 10, 10),

    Language('Kotlin', 11, 11),

    Language('Scala', 12, 12),

]

ides = [

    IDE('PyCharm', 1),

    IDE('Visual Studio', 2),

    IDE('Visual Studio Code', 3),

    IDE('IntelliJ IDEA', 4),

    IDE('Eclipse', 5),

    IDE('NetBeans', 6),

    IDE('Xcode', 7),

    IDE('Android Studio', 8),

    IDE('Rider', 9),

    IDE('CLion', 10),

    IDE('AppCode', 11),

    IDE('GoLand', 12),

]

language\_ide = [

    LanguageIDE(1, 1),

    LanguageIDE(2, 2),

    LanguageIDE(3, 2),

    LanguageIDE(4, 4),

    LanguageIDE(5, 3),

    LanguageIDE(6, 5),

    LanguageIDE(7, 1),

    LanguageIDE(8, 7),

    LanguageIDE(9, 12),

    LanguageIDE(10, 10),

    LanguageIDE(11, 8),

    LanguageIDE(12, 4),

]

# Language and IDE соединены соотношением один-ко-многим.

# Вывести все среды разработки, содержащие "Studio"

# и соответствующие им языки программирования, согласно language\_ide.

*def* task1():

    res = []    # Итог

    for ide in ides:

        mid\_res = []    # Промежуточный итог

        if "Studio" in ide.name:

            for lang\_ide in language\_ide:

                if lang\_ide.ide\_id == ide.id:

                    for lang in languages:

                        if lang.id == lang\_ide.language\_id:

                            mid\_res.append(lang.name)

            res.append([ide.name, mid\_res])

    return res

# Language and IDE соединены соотношением один-ко-многим.

# Выведите список IDE со средней сложностью Language в каждом IDE,

# отсортированный по средней сложности.

# Средняя сложность должна быть округлена до 2 знака после запятой

*def* task2():

    res = []

    for ide in ides:

        mid\_res = []

        sum = 0

        count = 0

        for lang\_ide in language\_ide:

            if lang\_ide.ide\_id == ide.id:

                for lang in languages:

                    if lang.id == lang\_ide.language\_id:

                        sum += lang.diificulity

                        count += 1

        try:

            mid\_res.append(ide.name)

            mid\_res.append(round(sum / count, 2))

            res.append(mid\_res)

        except ZeroDivisionError:

            pass

    return sorted(res, *key*=itemgetter(1), *reverse*=False)

# «IDE» и «Language» связаны соотношением многие-ко-многим.

# Выведите список всех language, у которых название начинается с буквы «S»,

# и названия их IDE

*def* task3():

    res = []

    for lang in languages:

        mid\_res = []

        if lang.name.startswith('S'):

            for lang\_ide in language\_ide:

                if lang\_ide.language\_id == lang.id:

                    for ide in ides:

                        if ide.id == lang\_ide.ide\_id:

                            mid\_res.append(ide.name)

            res.append([lang.name, mid\_res])

    return res

*def* main():

    print("Task 1: ")

    print(task1())

    print("Task 2: ")

    print(task2())

    print("Task 3: ")

    print(task3())

if \_\_name\_\_ == "\_\_main\_\_":

    main()

**Результаты работы программы**

