Отчёт по Рубежному контролю №1

Вариант предметной области: №23
Вариант запросов: D

Текст программы

Ниже представлены фотографии исходного кода программы.

rk1.py

rk1.py > LangSyntax

```python
from operator import itemgetter


class SyntaxConstruction:
    def __init__(self, id, name, complexity, lang_id):
        self.id = id
        self.name = name
        self.complexity = complexity
        self.lang_id = lang_id


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


class LangSyntax:
    def __init__(self, lang_id, constr_id):
        self.lang_id = lang_id
        self.constr_id = constr_id

langs = [
    ProgrammingLanguage(1, "Ada"),
    ProgrammingLanguage(2, "Assembly"),
    ProgrammingLanguage(3, "Python"),
    ProgrammingLanguage(4, "Java"),
    ProgrammingLanguage(5, "C++"),
]

constructions = [
    SyntaxConstruction(1, "condition", 2, 3),
    SyntaxConstruction(2, "iteration", 3, 3),
    SyntaxConstruction(3, "recursion", 5, 4),
    SyntaxConstruction(4, "lambda", 4, 3),
    SyntaxConstruction(5, "class definition", 3, 4),
    SyntaxConstruction(6, "function definition", 2, 1),
]

```

rk1.py

rk1.py > LangSyntax

```python
lang_syntax = [
    LangSyntax(1, 1),
    LangSyntax(3, 1),
    LangSyntax(4, 1),
    LangSyntax(3, 2),
    LangSyntax(4, 2),
    LangSyntax(3, 3),
    LangSyntax(4, 3),
    LangSyntax(3, 4),
    LangSyntax(4, 5),
    LangSyntax(1, 6),
]


def main():
    one_to_many = [
        (c.name, c.complexity, l.name)
        for l in langs
        for c in constructions
        if c.lang_id == l.id
    ]

    many_to_many_temp = [
        (l.name, ls.lang_id, ls.constr_id)
        for l in langs
        for ls in lang_syntax
        if l.id == ls.lang_id
    ]

    many_to_many = [
        (c.name, c.complexity, lang_name)
        for lang_name, lang_id, constr_id in many_to_many_temp
        for c in constructions
        if c.id == constr_id
    ]

```

rk1.py

rk1.py > LangSyntax

```python
def main():
    print("Задание D1")
    res_d1 = [
        (name, lang_name)
        for name, complexity, lang_name in one_to_many
        if name.endswith("ion")
    ]
    print(res_d1)

    print("\nЗадание D2")
    res_d2_unsorted = []
    for l in langs:
        l_constrs = [
            (name, complexity, lang_name)
            for name, complexity, lang_name in one_to_many
            if lang_name == l.name
        ]
        if l_constrs:
            complexities = [complexity for _, complexity, _ in l_constrs]
            avg_complexity = sum(complexities) / len(complexities)
            res_d2_unsorted.append((l.name, avg_complexity))

    res_d2 = sorted(res_d2_unsorted, key=itemgetter(1))
    print(res_d2)

    print("\nЗадание D3")
    res_d3 = {}
    for l in langs:
        if l.name.startswith("A"):
            l_constrs = [
                (name, complexity, lang_name)
                for name, complexity, lang_name in many_to_many
                if lang_name == l.name
            ]
            constr_names = [name for name, _, _ in l_constrs]
            res_d3[l.name] = constr_names
    print(res_d3)


if __name__ == "__main__":
    main()
```

Результаты работы программы

Вывод, полученный при запуске программы, приведён на фото ниже.

```
sejrandovletov@MacBook-Pro-Sejran rk % python3 rk1.py
Задание D1
[('function definition', 'Ada'), ('condition', 'Python'), ('iteration', 'Python'), ('recursion', 'Java'), ('class definition', 'J
ava')]

Задание D2
[('Ada', 2.0), ('Python', 3.0), ('Java', 4.0)]

Задание D3
{'Ada': ['condition', 'function definition'], 'Assembly': []}
```