

РК1 Сысоев Иван ИУ5-31Б Вариант 23

В рамках контрольной работы реализован программный код для обработки данных о синтаксических конструкциях и языках программирования. Основная задача заключается в создании классов для представления информации о конструкциях и языках, а также выполнения различных операций над этими данными в соответствии с условием задания.

Условие задания

1. Создать два класса данных, связанных отношениями один-ко-многим и многие-ко-многим. Классы должны содержать соответствующие идентификаторы и атрибуты в соответствии с предметной областью.
2. Создать списки объектов классов с тестовыми данными (3-5 записей) и настроить связи между ними.
3. Разработать три запроса для обработки данных:
 - Список объектов с их связями, отсортированный по одному из атрибутов.
 - Список объектов с суммарным количественным признаком, отсортированный по этому признаку.
 - Список объектов с фильтрацией по определённому условию.

Классы и их назначение

Класс Syntax

Представляет информацию о синтаксической конструкции.

Поля:

- `id` : ID конструкции (уникальный идентификатор).
- `name` : Название конструкции.
- `execution_time` : Время выполнения конструкции.
- `lang_id` : ID языка программирования, к которому относится конструкция.

Класс Language

Представляет информацию о языке программирования.

Поля:

- `id` : ID языка (уникальный идентификатор).
- `name` : Название языка.

Класс SyntaxLang

Реализует связь "многие-ко-многим" между синтаксическими конструкциями и языками программирования.

Поля:

- `syntax_id` : ID конструкции.
- `lang_id` : ID языка.

Данные

Языки программирования:

- Python
- Java
- C++
- C

Синтаксические конструкции:

- Функция
- Цикл do while
- Условный оператор if
- Цикл for
- Класс

Связь многие-ко-многим:

- Python: Функция, Условный оператор if, Цикл for, Класс
- Java: Функция, Условный оператор if, Цикл for, Класс
- C++: Функция, Цикл do while, Условный оператор if, Цикл for, Класс
- C: Функция, Условный оператор if, Цикл do while, Цикл for

Реализованные функции

1. Список конструкций по языкам программирования

Функция сортирует языки программирования по названию и выводит список синтаксических конструкций, принадлежащих каждому языку.

Пример вывода:

```
[('Класс', 0.05, 'Java'), ('Условный оператор if', 0.01, 'C++'), ('Функция', 0.04, 'Python'), ('Цикл do while', 0.01, 'C++'), ('Цикл for', 0.02, 'C')]
```

2. Список языков по суммарному времени выполнения конструкций

Функция вычисляет суммарное время выполнения конструкций для каждого языка программирования, сортирует языки по этому показателю и выводит результат.

Пример вывода:

```
[('Java', 0.05), ('Python', 0.04), ('C++', 0.02), ('C', 0.02)]
```

3. Список языков с конструкциями, содержащими "C" в названии

Функция находит языки программирования, в названии которых содержится "C", и выводит список синтаксических конструкций, связанных с ними.

Пример вывода:

```
{ 'C++': ['Функция', 'Цикл do while', 'Условный оператор if', 'Цикл for', 'Класс'], 'C': ['Функция', 'Цикл do while', 'Условный оператор if', 'Цикл for'] }
```

Выводы

В результате работы было реализовано:

- Моделирование данных о синтаксических конструкциях и языках программирования с использованием классов.
- Запросы для получения информации о конструкциях и языках.
- Использование сортировок и фильтрации для обработки данных.

Код успешно решает поставленные задачи и демонстрирует навыки работы с объектами и коллекциями в Python.