Сообщить об ошибке.

Общие операции с множествами set и frozenset



firecode.ru

РЕКЛАМА

Аутсорсинг разработки на Python - Django, DRF, Flask

Обслуживаем крупнейшие компании РФ: FinTech, MedTech, Retail, EdTech, AgroTech

Узнать больше

Справочник по языку Python3. / Общие операции с множествами set и frozenset

Здесь перечислены все доступные <u>операции с неизменяемыми множествами frozenset</u>, с их описанием и примерами использования в коде.

Дополнительно смотрите <u>описание типа frozenset</u> в языке программирования Python.

<u>Определение количества элементов в множестве в Python</u>

Oперация len(sets) вернет количество элементов в множестве sets (кардинальное число множества sets). Эта операция поддерживаются как неизменяемыми frozenset, так изменяемыми множествами set.

Проверка вхождения элемента в множество в Python

Операция x in sets проверяет наличие значения элемента x в множестве sets. Если значение x присутствует в множестве операция вернет True, если нет, то False. Операция x not in sets противоположна.

Meтод sets.isdisjoint() в Python, отсутствие элементов в множестве

Merog sets.isdisjoint() возвращает True, если множество sets не имеет общих элементов с итерируемым объектом other. Итерируемый объект other, это объект поддерживающий итерацию по своим элементам, может быть список, кортеж, другое множество

<u>Meтод sets.issubset() в Python, вхождение элементов в множество</u>

Meтод sets.issubset() позволяет проверить находится ли каждый элемент множества sets в последовательности other. Метод возвращает True, если множество sets **является подмножеством** итерируемого объекта other, если нет, т

<u>Проверка множества на правильное подмножество в Python</u>

Математический оператор < (меньше) позволяет проверить, является ли множество sets подходящим подмножеством другого множества other. Множество меньше другого тогда и только тогда, когда первое множество является правильным подмножеством второго

Metog sets.issuperset() в Python, вхождение элементов в множество

Merog sets.issuperset() позволяет проверить находится ли каждый элемент последовательности other в множестве sets. Метод возвращает True, если множество sets является надмножеством итерируемого объекта other, если нет, то вернет False.

Проверка множества на правильное надмножество в Python

Математический оператор > (больше) позволяет проверить, является ли множество sets правильным надмножеством другого множества other, то есть выполняются ли условия sets >= other и set != other.





Meтод sets.union() в Python, объединение множеств

Merog sets.union() позволяет объединить множество с двумя или более последовательностями поддерживающих итерирование. Метод возвращает новое множество с элементами из множества sets и элементами вставленными из всех итерируемых объектов *other.

Метод sets.intersection() в Python, пересечение множеств

Merog sets.intersection() позволяет найти пересечение множества с одной или более последовательностями поддерживающих итерирование. Метод возвращает новое множество с элементами, общими для множества sets и всех итерируемых объектов *other

Метод sets.difference() в Python, разность множеств

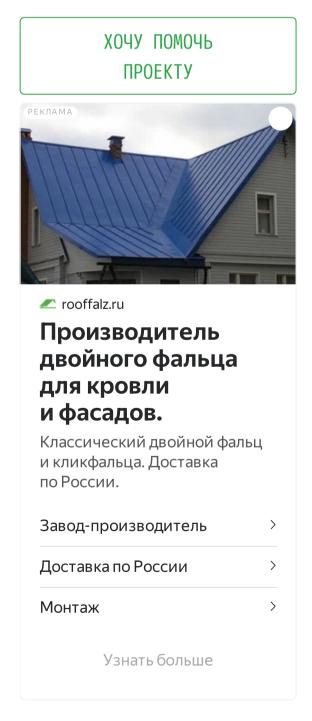
Meтод sets.difference() позволяет получить элементы множества, которых нет в одной или более последовательности поддерживающих итерирование. Метод возвращает новое множество с уникальными элементами множества sets, которых нет

<u>Метод sets.symmetric difference() в Python, симметричная разность</u>

Meтод sets.symmetric_difference() позволяет исключить из результата общие элементы для множества и последовательности, операцию еще называют симметричной разницей.

Meтод sets.copy() в Python, копия множества

Meтод sets.copy() вернет мелкую копию множества sets. Эта операция поддерживается как неизменяемым frozenset, так изменяемым множеством set.



DOCS-Python.ru[™], 2023 г.

(Внимание! При копировании материала ссылка на источник обязательна)

@docs_python_ru