

Советы и Трюки

Данный раздел создан для разработчиков, стремящихся упростить и ускорить свою работу, избегая рутинных задач. Здесь собраны практические советы и элегантные решения, которые помогут вам писать меньше кода, сделать вашу работу более эффективной и избавиться от необходимости «изобретать велосипед». Вы найдете здесь идеи для оптимизации разработки, примеры умных подходов к решению стандартных задач и творческие приемы, которые сделают ваш код более лаконичным и мощным. Этот раздел призван повысить вашу личную производительность, помочь сосредоточиться на творческой и значимой части работы, минимизируя рутинные и повторяющиеся процессы.

- [Использование оператора ?? для проверки доступа и сокращения количества кода](#)

Использование оператора ?? для проверки доступа и сокращения количества кода

Контекст

В разработке чат-ботов часто возникает потребность в управлении доступом к определенным разделам функционала на основе прав пользователя. Это может быть необходимо, например, для ограничения доступа к функциям инициации голосований в зависимости от роли пользователя в проекте.

Проблема

Традиционные подходы к решению этой задачи могут быть громоздкими и требовать написания обширного кода для проверки наличия и значений определенных ключей в объектах JSON. Это увеличивает сложность кода и его объем.

Решение: Использование Оператора ??

Оператор `??`, также известный как оператор нулевого слияния (nullish coalescing operator), предлагает элегантное решение для этой задачи. Он позволяет вернуть правую сторону выражения, если значение слева от `??` является `null` или `undefined`.

В контексте настройки меню чат-бота это означает, что можно легко проверить, имеет ли пользователь доступ к определенному пункту меню, не беспокоясь о том, что соответствующий элемент может отсутствовать в объекте.

Пример Использования

В данном примере мы используем `??` для управления отображением пунктов меню в зависимости от прав доступа пользователя:

```
// Получение списка проектов с правом инициации голосований
let projects = get_projects_for_voting_initiation(member_id)
memory.setJsonAttr("projects", projects)

// Возвращает массив, ключ - ID проекта, а значение - право доступа:
// {
//   "1": 0,
//   "2": 1
// }
```

Тогда в меню можно использовать следующий лаконичный синтаксис:

```
// Условия вывода пунктов меню
// Если projects[1] (Metabot Platform) равно true, показываем пункт меню
return memory.getJsonAttr("projects")[1] ?? false

// Если projects[2] (Metabot & Co.) равно true, показываем пункт меню
return memory.getJsonAttr("projects")[2] ?? false
```

Пример:

MENU

Set up here the menu items that the bot will send to the user.

The code is the text that the user must send to get to the desired section of the menu.

The menu is optional. If the menu is not there, the bot will show the user the menu from the section above (except Start).

CODE	CAPTION	SCRIPT	DISPLAY CONDITION
1	Metabot Platform	Metabot Platform	<code>return memory.getJsonAttr("projects")[1] ?? false</code>
2	Metabot & Co.	Metabot & Co.	<code>return memory.getJsonAttr("projects")[2] ?? false</code>
3	CoCreatives	CoCreatives	<code>return memory.getJsonAttr("projects")[4] ?? false</code>
99	 Отмена	 Главное меню	

Преимущества

- Упрощение Кода:** Уменьшается количество кода, необходимого для проверки доступности пунктов меню.
- Надежность:** Снижается вероятность ошибок, связанных с неопределенными или нулевыми значениями.
- Лучшая Читаемость:** Код становится более понятным и легче поддерживаемым.