

Задание: Редактор для поиска и замены текста.

Описание:

Создать одностраничное веб-приложение, которое позволяет пользователю вводить текст, а затем находить и заменять определённые фрагменты с помощью регулярных выражений. Это задание демонстрирует мощь регулярных выражений для обработки и манипуляции текстом.

Требования:

HTML:

- `textarea` для ввода основного текста.
- Два поля ввода (`input type="text"`) — одно для шаблона поиска, другое для текста замены.
- Две кнопки: "Найти все" и "Заменить все".
- Элемент для отображения результатов поиска (например, `div` или `p`), где будет показано количество совпадений и их список.

CSS:

- Оформить все элементы интерфейса, чтобы приложение выглядело аккуратно и удобно.
- (По желанию) Стилизовать подсветку найденных совпадений непосредственно в `textarea` (это более продвинутая задача, её можно реализовать, используя JavaScript для динамического создания элементов с подсветкой).

JavaScript:

- Обработчик кнопки "Найти все":

- Получить текст из `textarea` и шаблон из поля поиска.
- Создать новый объект `RegExp` из строки-шаблона, добавив флаг `g` (глобальный поиск).
- Использовать метод `.matchAll()` для нахождения всех совпадений и их позиций. Если браузер не поддерживает `matchAll`, использовать `.match(регулярка)`.
- Вывести количество найденных совпадений и их список в отдельном блоке на странице.
- Обработчик кнопки "Заменить все":
- Получить текст из `textarea`, шаблон поиска и текст для замены.
- Создать регулярное выражение с флагом `g`.
- Использовать метод `.replace(регулярка, 'замена')` для замены всех совпадений.
- Обновить текст в `textarea` на новый, изменённый.

Дополнительно (по желанию):

- Режим "Поиск по умолчанию": Добавить кнопки-ярлыки, которые автоматически подставляют в поле поиска популярные регулярные выражения, например:
- Email: Находит все email-адреса.
- URL: Находит все URL-адреса.
- Телефон: Находит все номера телефонов в формате +7 (XXX) XXX-XX-XX.

Шпаргалка по работе с регулярными выражениями

`const pattern = /шаблон/;` Объявление регулярного выражения.

`const pattern = new RegExp(строка, 'g');` Создание регулярного выражения из строки, что позволяет динамически добавлять флаги, например `g`.

`string.matchAll(pattern)`: Возвращает итератор со всеми совпадениями, включая группы захвата. Идеально для поиска всех совпадений.

`string.replace(pattern, replacement)`: Заменяет все совпадения. Используйте `g` флаг, чтобы заменить все, а не только первое.