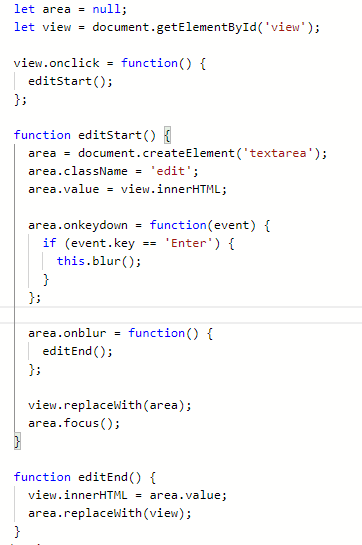
**[Редактируемый div](https://learn.javascript.ru/focus-blur" \l "redaktiruemyy-div)**

важность: 5

Создайте <div>, который превращается в <textarea>, если на него кликнуть.

<textarea> позволяет редактировать HTML в элементе <div>.

Когда пользователь нажимает Enter или переводит фокус, <textarea> превращается обратно в <div>, и его содержимое становится HTML-кодом в <div>.



### [Редактирование TD по клику](https://learn.javascript.ru/focus-blur" \l "redaktirovanie-td-po-kliku)

важность: 5

Сделайте ячейки таблицы редактируемыми по клику.

* По клику – ячейка должна стать «редактируемой» (textarea появляется внутри), мы можем изменять HTML. Изменение размера ячейки должно быть отключено.
* Кнопки OK и ОТМЕНА появляются ниже ячейки и, соответственно, завершают/отменяют редактирование.
* Только одну ячейку можно редактировать за один раз. Пока <td> в «режиме редактирования», клики по другим ячейкам игнорируются.
* Таблица может иметь множество ячеек. Используйте делегирование событий.
* let table = document.getElementById('bagua-table');
* let textareaEl = null;
* table.addEventListener('click', function (event) {
* if (textareaEl) return;                                      /\* уже редактируется - не реагировать на клики на table \*/
* textareaEl = document.createElement('textarea');
* textareaEl.classList.add("edit");
* let buttonOK = document.createElement('button');
* buttonOK.classList.add("ok");
* buttonOK.innerHTML = 'OK';
* let buttonRes = document.createElement('button');
* buttonRes.classList.add("res");
* buttonRes.innerHTML = 'Отмена';
* let td = event.target.closest('TD');
* if (!td || !table.contains(td)) return;                           /\* если клик не внутри td \*/
* textareaEl.value = td.innerHTML;
* textareaEl.style.width = td.clientWidth + 'px';                /\* подгоняем габаритьі textarea\*/
* textareaEl.style.height = td.clientHeight + 'px';
* td.replaceWith(textareaEl);                                     /\* textarea  вместо td \*/
* textareaEl.focus();
* document.body.append(buttonOK);
* buttonOK.style.top = textareaEl.getBoundingClientRect().bottom - 1 + 'px';  /\* buttonOK подвяжем к textarea \*/
* buttonOK.style.left = textareaEl.getBoundingClientRect().left + textareaEl.clientWidth / 2 - buttonOK.clientWidth + 'px';
* document.body.append(buttonRes);
* buttonRes.style.top = textareaEl.getBoundingClientRect().bottom - 1 + 'px'; /\* buttonRes подвяжем к textarea и buttonOK \*/
* buttonRes.style.left = buttonOK.getBoundingClientRect().right + 'px';
* buttonOK.addEventListener('click', function () {     /\* принимаем изменения, собьітие на кнопке (на table оно отсекается)\*/
* td.innerHTML = textareaEl.value;
* textareaEl.replaceWith(td);
* textareaEl = null;                 /\* удаляем текущий textareaEl для возм редактирования новой td \*/
* buttonOK.remove();
* buttonRes.remove();
* });
* buttonRes.onclick = () => {
* textareaEl.replaceWith(td);
* textareaEl = null;
* buttonOK.remove();
* buttonRes.remove();
* };
* });

Css:

.edit {

            position: relative;

            text-align: left;

            white-space: nowrap;

            margin: 0;

            padding: 0;

        }

        .ok {

            position: absolute;

            height: 25px;

            width: 55px;

            background-color: rgb(60, 236, 60);

            cursor: pointer;

        }

        .res {

            position: absolute;

            height: 25px;

            width: 60px;

            background-color: rgb(250, 3, 3);

            cursor: pointer;

        }

### [Мышь, управляемая клавиатурой](https://learn.javascript.ru/focus-blur" \l "mysh-upravlyaemaya-klaviaturoy)

важность: 4

Установите фокус на мышь. Затем используйте клавиши со стрелками, чтобы её двигать:

[Демо в новом окне](https://ru.js.cx/task/keyboard-mouse/solution/)

P.S. Не добавляйте обработчики никуда, кроме элемента #mouse.

P.P.S. Не изменяйте HTML/CSS, подход должен быть общим и работать с любым элементом.

