

Fullstack-разработка

Vue

Поля ввода

## Цели урока:

1. Изучим специальную директиву для полей ввода
2. Познаем условный рендеринг

# Разбор ДЗ

## Лампочка

# Теория

## Текстовые поля

**Директива** – это ключевое слово, которое используется в шаблоне и выполняет особое действие.

**Уже знаем:**

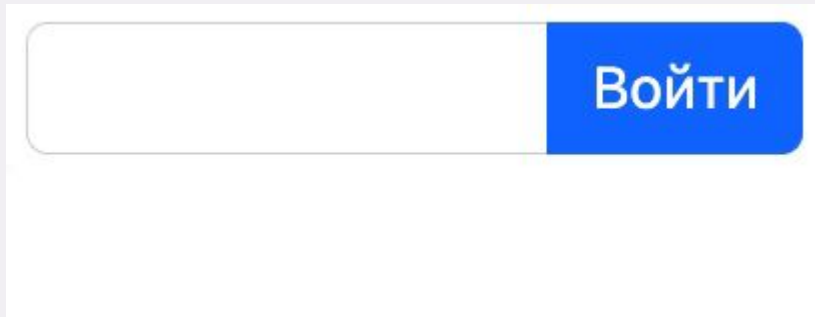
1. **v-on** привязывает событие  
@click, @input, @keydown
2. **v-bind** привязывает атрибут  
:class, :style, :src

**Директива** – это ключевое слово, которое используется в шаблоне и выполняет особое действие.

**Изучим сегодня:**

- 3. **v-model** работает с полями ввода
- 4. **v-if** создаёт условия в шаблоне

## Знакомая задача для JS?



A login form consisting of a white rectangular box with rounded corners. Inside the box, on the left, is a text input field with a light gray border and rounded corners. To the right of the input field is a blue button with rounded corners and the white text 'Войти'.

В Vue она решается ещё проще!

# input

У поля ввода много разных атрибутов, давайте вспомним, за что отвечает каждый:

```
<input  
  type="text"      // разный тип  
  id="username"   // связь с лейблом  
  name="username" // отправка на сервер  
  value="Админ"   // введённое значение  
>
```

# v-model

Во Vue есть специальная **директива** для работы со **значением** поля ввода.

```
<input  
  type="text"  
  id="username"  
  name="username"  
  v-model="username"  
>
```

```
data() {  
  return {  
    username: 'Админ'  
  }  
}
```

# v-model

**Data-свойство** обновляется каждый раз, когда меняется **значение** в поле ввода.

```
<input type="text" id="username"  
       name="username"  
       v-model="username" >
```

```
{{ username }}
```

Войти

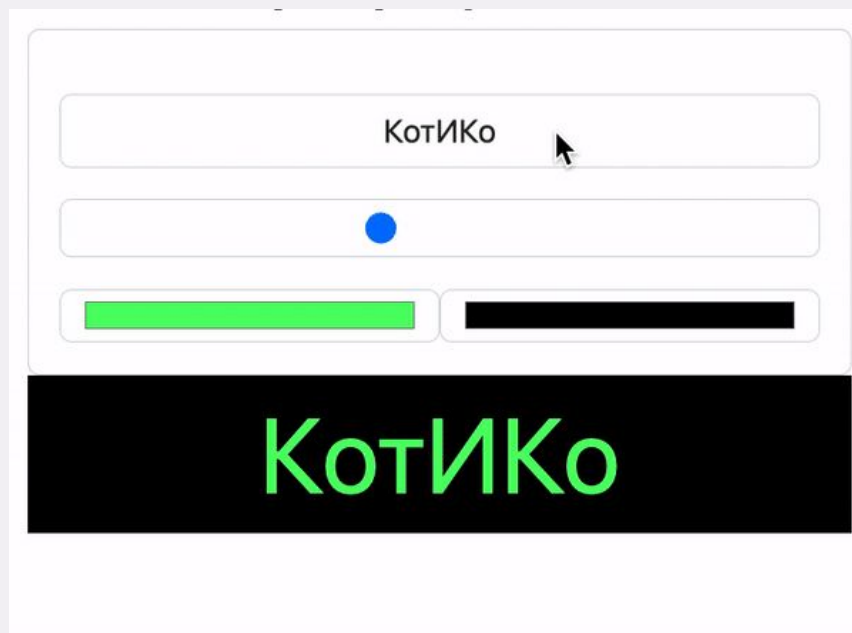
Привет, !

# v-model

**Значение** обновляется каждый раз, когда  
меняется **data-свойство**.

```
methods: {  
  hello() {  
    this.username = '(-▽-)ノ';  
  }  
}
```

Всё! Попробуй применить  
новое знание на практике:



A screenshot of a web form. At the top is a text input field containing the text "КотиКо". Below it is a progress bar with a blue dot in the center. At the bottom is a large black button with the text "КотиКо" in green.

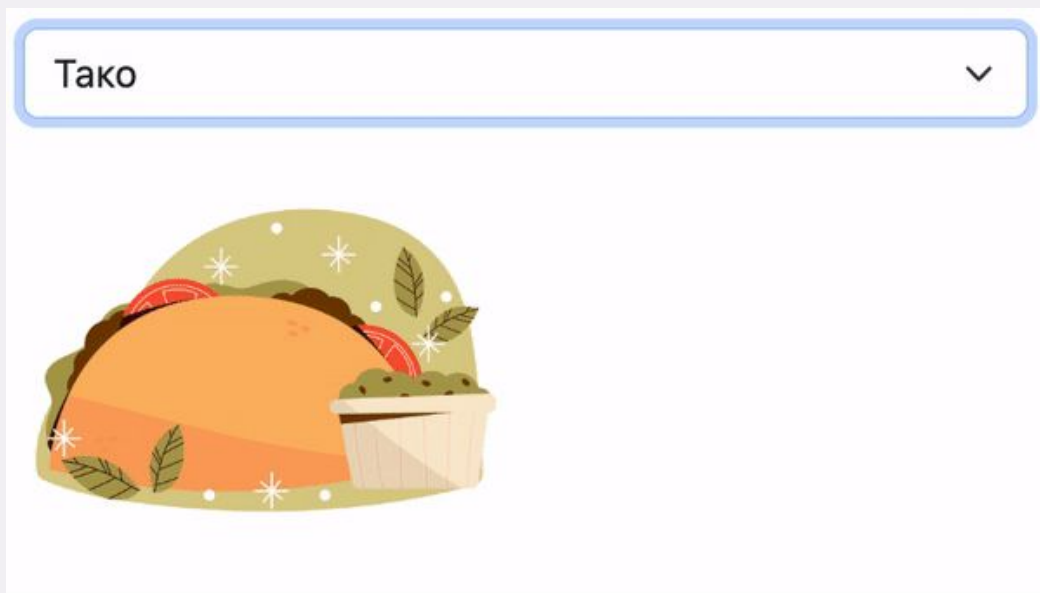
# Практика

## Редактор текста 2

# Теория

## Поля с выбором

## select и v-model



# select

В **data-свойство** будет подставляться **value** выбранного option.

```
<select v-model="image">
  <option value="taco">Тако</option>
  <option value="ramen">Памен</option>
  <option value="hotdog">Хотдог</option>
</select>
```

## select

Внутри атрибута **v-bind** можно использовать выражение, которое составит правильный путь до картинки:

```

```

## checkbox и v-model

☒ Включить уведомления

Теперь вы не пропустите специальные закрытые распродажи!

Проверьте вашу почту: первое письмо уже там.

# checkbox

У чекбокса есть только два состояния:  
**выбран** и **не выбран**.

Поэтому его **v-model** – это всегда значение  
с типом `boolean`.

```
<input class="form-check-input"
  v-model="isVisible"
  type="checkbox">
```

```
data() {
  return {
    isVisible: false
  }
}
```

## v-if

Чтобы скрывать и показывать элементы в шаблоне, используйте директиву **v-if**:

```
<div v-if="isVisible">
```

*Видно, если isVisible == true*

```
</div>
```

```
<div v-if="!isVisible">
```

*Видно, если isVisible == false*

```
</div>
```

```
<div v-if="username == 'Админ'">
```

*Видно, если username равен строке Админ*

```
</div>
```

# Практика

Редактор текста 3

Вход на сайт

# Теория

## Радиокнопки

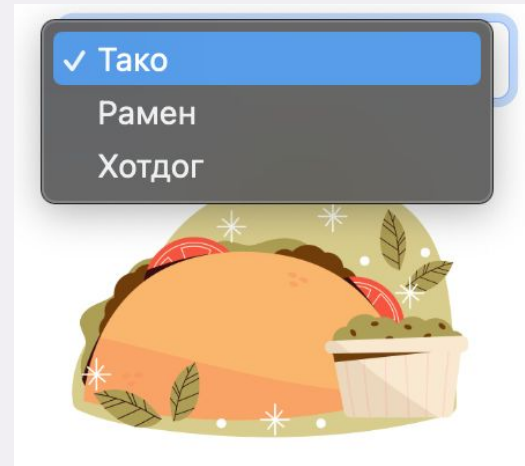
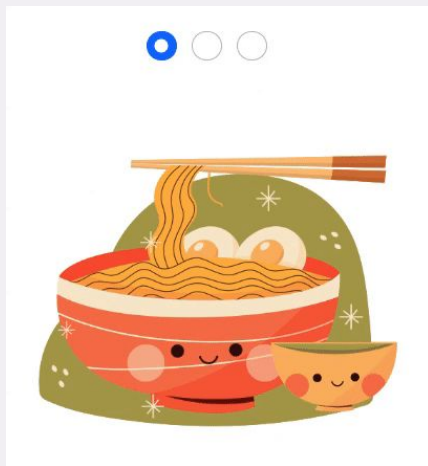
Радиокнопки – это набор элементов, среди которых можно выбрать только один.

### Рейтинг фильма



*Звёздочки на Кинопоиске*

Радиокнопки по смыслу  
похожи на **select**,  
но иногда бывают удобнее  
для пользователя.



# radio

Группа **радиокнопок** имеет общий **v-model** и разные значения **value**.

```
<div class="form-check">
  <input class="form-check-input"
    type="radio"
    v-model="image" value="ramen">
</div>

<div class="form-check">
  <input class="form-check-input"
    type="radio"
    v-model="image" value="taco">
</div>
```

# Практика

Какой ты овощ?

## Цели урока:

- ✓ Изучили работу с полями ввода:  
**v-model**
- ✓ Познакомились с условным рендерингом:  
**v-if**

## Поля ввода

1. Управляются через `v-model`  
`v-model="username"`
2. Для чекбоксов `v-model`  
всегда типа `boolean`  
`v-model="isHidden"`
3. Для группы радиокнопок  
`v-model` будет одинаковым

## Условный рендеринг

Директива `v-if` скрывает или показывает блок в зависимости от условия

```
<div v-if="!isLoading">  
  Идёт загрузка...  
</div>
```

## Домашнее задание