

Mit einem Keyboard Event kann ein Tastendruck ermittelt werden. Die Events können als Attribut in einem HTML Element definiert werden bzw. dynamisch mit einem Eventlistener. Ein Objekt wäre z. B. `var objekt = document.querySelector("input");`

**onkeydown** feuert sobald eine Taste heruntergedrückt wird

**JS** `objekt.addEventListener("keydown", meineFunktion);`

**HTML** `<div onkeydown="meineFunktion()">`

**onkeyup** feuert sobald eine Taste losgelassen wird

**JS** `objekt.addEventListener("keyup", meineFunktion);`

**HTML** `<div onkeyup="meineFunktion()">`

**onkeypress** wenn Taste heruntergedrückt und festgehalten wird.

**JS** `objekt.addEventListener("keypress", meineFunktion);`

**HTML** `<div onkeypress="meineFunktion()">`

### Wertübergabe mit event

Startet man eine Funktion mit der Wertübergabe `event`, dann wird der Tastendruck an die Funktion übergeben. Mit `event.key` wird der Tastendruck ausgewertet.

**JS** `var meineTaste = event.key;`

```
<input type="text" onKeyDown="welcheTaste(event);" >

<script>
  function welcheTaste(event) {
    var meineTaste = event.key;
    alert(meineTaste);
  }
</script>
```

### event.keyCode Eigenschaft

**JS** `event.keyCode` ← Gibt den Code der Taste zurück.  
z. B.: a = 97, b = 98, Enter = 13

```
function welcheTaste(event) {
  if(event.keyCode == 13) {alert("Enter wurde gedrückt");}}
```



Weitere Eigenschaften die einen Unicode zurückgeben sind:  
`event.charCode` oder `event.which`.

*Beachte:* `event.which` wird nicht vom Internet Explorer unterstützt und `event.keyCode` funktioniert nicht als `onkeypress` Event im Firefox.

Deshalb gibt es eine Cross-Browser-Solution in der einfach beide Eigenschaften vereinbart werden (abhängig vom Browser):

```
var meineTaste = event.which || event.keyCode;
```