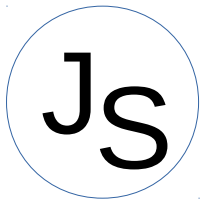


- Событийно-ориентированное программирование
- Добавление обработчиков событий
- События браузера
- События документа
- События формы
- События клавиатуры
- События мыши

Событийно-ориентированное программирование

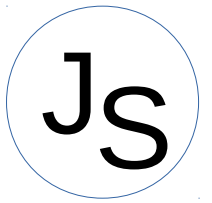


- Слабые связи в коде
- Гибкость, простота адаптации
- Бесшовная связь с сервером через события



- Трудно отлаживать
- Трудно сопровождать

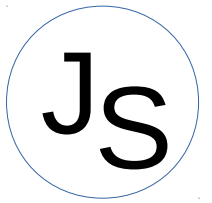
Событийно-ориентированное программирование



```
var Foo = function() {  
    this.lorem = function()  
    {  
        alert("HELLO FOO");  
    }  
};
```

```
var Bar = function() {  
    var foo = new Foo();  
    foo.lorem();  
};
```

Событийно-ориентированное программирование



```
var Foo = function() {  
  
    myEventSystem.on("lorem.event", lorem);  
  
    function lorem()  
    {  
        alert("HELLO FOO");  
    }  
}
```

```
var Bar = function() {  
    var foo = new Foo();  
    myEventSystem.send("lorem.event");  
};
```

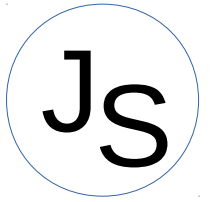
Добавление обработчиков



```
<input
  id="b1"
  type="button"
  value="Click me"
  onclick="alert('Thanks!');"
/>
```

```
<input id="myElement" type="button" value="Жми"/>
<script>
  document.getElementById('myElement').onclick =
  function() {
    alert('Thanks');
  }
</script>
```

Добавление обработчиков



```
var element = document.getElementById("myElement");  
var handler = function() { alert('Thanks'); }
```

```
element.addEventListener("click", handler);
```

```
element.removeEventListener("click", handler);
```

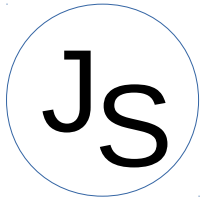
События браузера



- online
- offline
- error
- open
- popstate
- load
- unload
- beforeunload

```
addEventListener(  
    "beforeunload",  
    ()=>confirm('Вы уверены?')  
);
```

События документа



- DomContentLoaded

```
document.addEventListener(  
  "DOMContentLoaded",  
  function() {  
    document.body.addEventListener('click', ...);  
  }  
);
```


События формы



- submit
- reset
- focus
- blur
- change
- input
- select

```
function areYouSure(e) {  
    if (!confirm('Are you sure?')) {  
        e.preventDefault();           //return false;  
    }  
}  
  
myForm.addEventListener("submit", areYouSure);
```

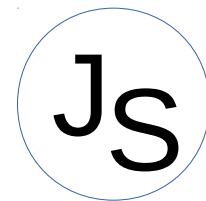
События клавиатуры



- keypress
- keydown
- keyup

```
element.addEventListener('keydown', function(e) {  
    if(  
        e.ctrlKey &&  
        String.fromCharCode(e.keyCode) == 'W'  
    ) {  
        alert('You pressed Ctrl-W');  
    }  
})
```

События мыши

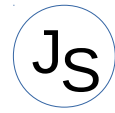


- click
- dblclick
- mousedown
- mouseup
- mouseenter
- mouseleave
- mousemove
- mouseover
- mouseout

```
element.addEventListener(  
  'mouseover',  
  function(e) {  
    e.target.style.borderColor  
      = 'blue';  
    e.stopPropagation();  
  }  
)
```

- Событийно-ориентированное программирование
- Добавление обработчиков событий
- События браузера
- События документа
- События формы
- События клавиатуры
- События мыши

Событийно-ориентированное программирование

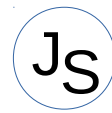


- Слабые связи в коде
- Гибкость, простота адаптации
- Бесшовная связь с сервером через события



- Трудно отлаживать
- Трудно сопровождать

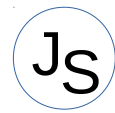
Событийно-ориентированное программирование



```
var Foo = function() {  
    this.lorem = function()  
    {  
        alert("HELLO FOO");  
    }  
};
```

```
var Bar = function() {  
    var foo = new Foo();  
    foo.lorem();  
};
```

Событийно-ориентированное программирование



```
var Foo = function(){  
  
    myEventSystem.on("lorem.event", lorem);  
  
    function lorem()  
    {  
        alert("HELLO FOO");  
    }  
}
```

```
var Bar = function(){  
    var foo = new Foo();  
    myEventSystem.send("lorem.event");  
};
```

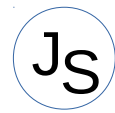
Добавление обработчиков



```
<input
  id="b1"
  type="button"
  value="Click me"
  onclick="alert('Thanks!');"
/>
```

```
<input id="myElement" type="button" value="Жми"/>
<script>
  document.getElementById('myElement').onclick =
  function() {
    alert('Thanks');
  }
</script>
```


Добавление обработчиков

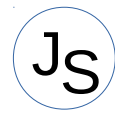


```
var element = document.getElementById("myElement");  
var handler = function(){ alert('Thanks'); }
```

```
element.addEventListener("click", handler);
```

```
element.removeEventListener("click", handler);
```

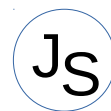
События браузера



- online
- offline
- error
- open
- popstate
- load
- unload
- beforeunload

```
addEventListener(  
  "beforeunload",  
  ()=>confirm('Вы уверены?')  
);
```

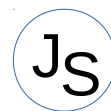
События документа



- DomContentLoaded

```
document.addEventListener(  
  "DomContentLoaded",  
  function() {  
    document.body.addEventListener('click', ...);  
  }  
);
```

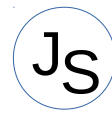
События формы



- submit
- reset
- focus
- blur
- change
- input
- select

```
function areYouSure(e) {  
    if (!confirm('Are you sure?')) {  
        e.preventDefault();           //return false;  
    }  
}  
  
myForm.addEventListener("submit", areYouSure);
```

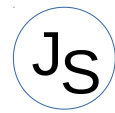
События клавиатуры



- keypress
- keydown
- keyup

```
element.addEventListener('keydown', function(e) {  
  if(  
    e.ctrlKey &&  
    String.fromCharCode(e.keyCode) == 'W'  
  ) {  
    alert('You pressed Ctrl-W');  
  }  
})
```

События мыши



- click
- dblclick
- mousedown
- mouseup
- mouseenter
- mouseleave
- mousemove
- mouseover
- mouseout

```
element.addEventListener(  
  'mouseover',  
  function(e) {  
    e.target.style.borderColor  
      = 'blue';  
    e.stopPropagation();  
  }  
)
```