

Język JavaScript

Inne mechanizmy

Tablice

Wyrażenia
regularne

...

Tablice – dwa sposoby patrzenia

- tablice jako **stosy**

```
var cs = ['red', 'blue'];
cs.push('brown');
cs[3] = 'black';
console.log(cs.length); // 4
var c = cs.pop();
console.log(c);          // 'black'
```

- tablice jako **kolejki**

```
var cs = ['red'];
cs.unshift('blue', 'green');
console.log(cs.length); // 3
var c = cs.shift();
console.log(c);          // 'red'
console.log(cs.length); // 2
```

Tablice – zmiana kolejności elementów

```
var tab = [0, 1, 5, 10, 15];
tab.reverse();
console.log(tab); // [15, 10, 5, 1, 0]
tab.sort();
console.log(tab); // [0, 1, 10, 15, 5]

var compare = function (a, b) {
  if (a < b) {
    return -1;
  } else if (a > b) {
    return 1;
  } else {
    return 0;
  }
};

tab.sort(compare);
console.log(tab); // [0, 1, 5, 10, 15]
```

Tablice – ciekawsze z pozostałych metod

```
var tab = [15, 10, 5, 1, 0];
var t1 = tab.filter(function (a) { return a > 5; });
console.log(tab); // [15, 10, 5, 1, 0]
console.log(t1); // [15, 10]

var t2 = tab.map(function (a) { return a * 2; });
console.log(t2); // [30, 20, 10, 2, 0]

var t3 = tab.reduce(function (prev, curr) {
  return prev + curr;
});
console.log(t3); // 31

var tab2 = ['s', 'i', 'v', 'l'];
var t4 = tab2.reduceRight(function (prev, curr) {
  return prev + curr;
}, 'E');
console.log(t4); // Elvis
```

Wyrażenia regularne

```
var txt = "cat, bat, sat, fat";
var pat = /.at/g;
var mat = pat.exec(txt)

console.log(mat.index); // 0
console.log(mat[0]); // cat
console.log(pat.lastIndex); // 3

mat = pat.exec(txt)

// Jaki będzie wynik?
console.log(mat.index); // 5
console.log(mat[0]); // bat
console.log(pat.lastIndex); // 8
```

Wyrażenia regularne

```
var txt = "cat, bat, sat, fat";
var pat = /.at/;
var mat = pat.exec(txt)

console.log(mat.index); // 0
console.log(mat[0]); // cat
console.log(pat.lastIndex); // 0

mat = pat.exec(txt)

// Jaki będzie wynik?
console.log(mat.index); // 0
console.log(mat[0]); // cat
console.log(pat.lastIndex); // 0
```

Wyrażenia regularne

```
var txt = "cat, bat, sat, fat";
var pat = /.at/g;
var mat = pat.exec(txt)

console.log(mat.index); // 0
console.log(mat[0]); // cat
console.log(pat.lastIndex); // 3

mat = pat.exec(txt)

// Jaki będzie wynik?
console.log(mat.index); // ?
console.log(mat[0]); // ?
console.log(pat.lastIndex); // ?
```

Wyrażenia regularne

```
var txt = "cat, bat, sat, fat";
var pat = /.at/g;
var dispMatch = function () {
  console.log("tekst: " + RegExp.input + "");
  console.log("dopasowanie: " + RegExp.lastMatch + "");
  console.log("tekst przed: " + RegExp.leftContext + "");
  console.log("tekst za: " + RegExp.rightContext + "");
  console.log("ostatnia grupa: " + RegExp.lastParen + "");
};

if (pat.test(txt)) {
  dispMatch();
}
// Jaki będzie wynik?
```

tekst: 'cat, bat, sat, fat'

dopasowanie: 'cat'

tekst przed: ''

tekst za: ', bat, sat, fat'

ostatnia grupa: ''



Małe ćwiczonko-zagadka :)

Wyrażenia regularne

```
var txt = "Dzisiaj mamy 2013-03-12 w „zapisie komputerowym.”;
var pat = /(\d{4})-(\d{2})-(\d{2})/;
var dispGroups = function () {
  var i;
  for(i = 1; i <= 9; i += 1) {
    console.log("$" + i + ": " + RegExp["$" + i] + "");
  }
};

if (txt.match(pat)) {
  dispGroups();
}
// Jaki będzie wynik?
```

\$1: '2013'

\$2: '03'

\$3: '12'

\$4: ''

... ''

\$9: ''



Wyrażenia regularne w JS

```
var template =
  '<table border="{border}">' +
  ' <tr><td>{first}</td><td>{last}</td></tr>' +
  '</table>';
```

```
var data = {
  first: "Jan",
  last: "Kowalski",
  border: "2"
};
```

```
template.supplant(data)
```



```
<table border="2">
  <tr><td>Jan</td><td>Kowalski</td></tr>
</table>
```

Jak to zdefiniować?