

Neden JavaScript

M Akif Eyler
TAI, Şubat 2020

Kullandığım Diller

- | | | |
|--------|----------------|------------------|
| • 1971 | Mainframe | Fortran |
| • 1975 | Terminal | Fortran ⇒ Pascal |
| • 1985 | IBM PC | Turbo Pascal |
| • 1990 | Macintosh | Object Pascal |
| • 2002 | Diz üstü | Java |
| • 2018 | Mobil cihazlar | JavaScript |

1971 Oda içi -- Mainframe



FORTRAN

Program listing

```
1      K=1
2      6      IF (K.EQ.11) GO TO 8
3          READ,I,J
4          IF (J.GT.I) GO TO 65
5          GO TO 66
6      65      WRITE(6,6002)J,I
7      6002     FORMAT(' ',I3,' IS GREATER THAN ',
8          K=K+1
9          GO TO 6
10     66      WRITE(6,6001)I,J
11     6001     FORMAT(' ',I3,' IS GREATER THAN ',
12     K=K+1
13     GO TO 6
14     8      CALL EXIT
```

1985 Mac & PC



Neden JavaScript

- her platformda çalışıyor
- daha kolay ve daha eğlenceli
- web tasarımında vazgeçilmez

<https://eyler.blogspot.com/2018/06/neden-javascript.html>

Name and Value

```
let user = 'Ali'           //string
let count = 234            //number
let time = new Date()      //object

user.length                //property
3

user.toLowerCase()         //method
"ali"

count.toFixed(2)
"234.00"

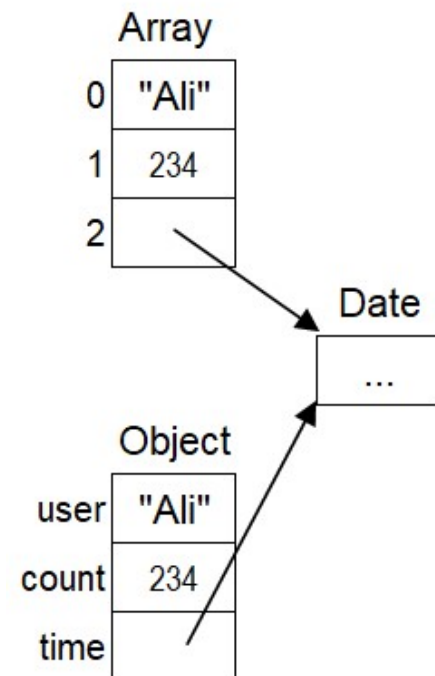
time.getFullYear()
2020

time.toString()
"Wed Feb 05 2020"
```

Array and Object

```
a = [user, count, time]
["Ali", 234, Wed Feb 05 2020 09:13:17]
a.length -> 33
a.indexOf(234) -> 1
a.filter(x => typeof x == 'string')
["Ali"]
```

```
p = {user, count, time}
{user: "Ali", count: 234, time: Wed ...}
p.user -> "Ali"
p.count -> 234
```



Array, String, JSON

a

```
["Ali", 234, Wed Feb 05 2020 09:13:17]
```

```
s = a.join(', ')
```

```
"Ali, 234, Wed Feb 05 2020 09:13:17"
```

```
b = s.split(', ')
```

```
["Ali", "234", "Wed Feb 05 2020 09:13:17"]
```

```
a == b -> false
```

```
b.join(', ') == s -> true
```

```
JSON.stringify(a)
```

```
'["Ali",234,"2020-02-05T06:13:17.026Z"]'
```

Array of Objects

```
let map = new Map()
for (let t of data) if (t.user == U) {
  let [id, txt] = t.topic.split('_')
  map.set(Number(id), [txt, t.marks])
} //input data:  { time, user, topic, marks }
                  akif  23_mvc  C

for (let id of map.keys()) {
  let [txt, mrk] = map.get(id)
  process(id, txt, mrk)
} //output data:  { id, text, complete }
                  23    mvc    true
```

Tools

Chrome



GitHub



GitKraken



VS Code – Live Server extension

JavaScript Örnekleri: <https://maeyler.github.io/JS/>

Eloquent JavaScript: <http://eloquentjavascript.net/>

What Web Can Do Today: <https://whatwebcando.today/>

Simple Cloud Storage

Forms to Sheets:

<https://maeyler.github.io/JS/data/>

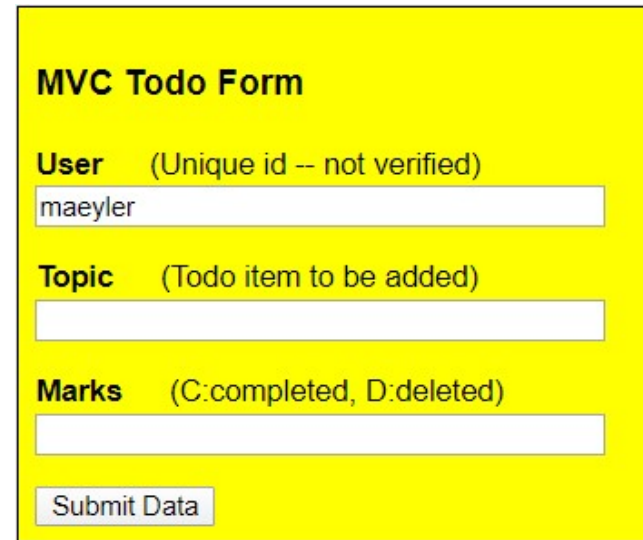
MVC application:

<https://github.com/taniarascia/mvc/>

MVC Todo App:

<https://maeyler.github.io/mvc/>

<https://maeyler.github.io/>

A screenshot of a web form titled "MVC Todo Form" with a yellow background. The form contains three input fields: "User" with the value "maeyler", "Topic", and "Marks". Each field has a label and a description in parentheses. At the bottom is a "Submit Data" button.

MVC Todo Form

User (Unique id -- not verified)
maeyler

Topic (Todo item to be added)

Marks (C:completed, D:deleted)

Submit Data