

# ÖDEV 2

Ahmet Yusuf Birdir  
21360859026

## process.argv

- Process.argv, bir Node.js programına dışarıdan parametre vermek için kullanılır. İşleyişi yandaki gibidir.

```
JS h2u1.js
```

```
1 console.log(process.argv)
```

```
PS C:\Users\abird\OneDrive\Masaüstü\Node_JS_lab\Hafta 2> node .\h2u1.js arg1  
[  
  'C:\\Program Files\\nodejs\\node.exe',  
  'C:\\Users\\abird\\OneDrive\\Masaüstü\\Node_JS_lab\\Hafta 2\\h2u1.js',  
  'arg1'  
]
```

# Yargs.argv

- yargs, Node.js için komut satırı argümanlarını daha kolay ve esnek şekilde işlemek için kullanılan bir kütüphanedir. Yandaki şekilde process.argv ile farkı gösterilmiştir.

```
PS C:\Users\abird\OneDrive\Masaüstü\Node_JS_lab\Hafta 2> node .\h2u2.js arg1
[
  'C:\\Program Files\\nodejs\\node.exe',
  'C:\\Users\\abird\\OneDrive\\Masaüstü\\Node_JS_lab\\Hafta 2\\h2u2.js',
  'arg1'
]
{ _: [ 'arg1' ], '$0': 'h2u2.js' } ← yargs.argv
PS C:\Users\abird\OneDrive\Masaüstü\Node_JS_lab\Hafta 2> 
```

```
JS h2u2.js > ...
1  const yargs = require("yargs")
2
3  console.log(process.argv)
4
5  console.log(yargs.argv)
```

# yargs.command ve özellikleri

- ▶ 1-) command: Komutun ismini tanımlar.
- ▶ 2-) describe: Komutun ne yapacağı ile ilgili açıklamadır.
- ▶ 3-) builder: Komut için gerekli olan argümanların tanımlandığı bloktur.
  - ▶ a-) title: Komutun başlığını belirler.
  - ▶ b-) body: Komutun başlığını belirler.
  - ▶ c-) handler: Komut çalıştığında çalışacak olan kodu içerir.

```
JS h2u3.js > ...
1  const yargs = require("yargs")
2
3  yargs.command({
4    command: 'ekle',
5    describe: 'yeni not ekler',
6    builder: {
7      title: {
8        describe: 'Başlık1',
9        demandOption: true,
10       type: 'string'
11      },
12      body: {
13        describe: 'gövde',
14        demandOption: true,
15        type: 'string'
16      }
17    },
18    handler: function(argv){
19      console.log(argv.title)
20      console.log(argv.body)
21    }
22  })
23
24  console.log(yargs.argv)
```

► İşte çalışma şekilleri:

```
PS C:\Users\abird\OneDrive\Masaüstü\Node_JS_lab\Hafta 2> node h2u3.js --help
h2u3.js [command]
```

Commands:

h2u3.js ekle      yeni not ekler

```
PS C:\Users\abird\OneDrive\Masaüstü\Node_JS_lab\Hafta 2> node h2u3.js ekle --help
h2u3.js ekle
```

yeni not ekler

Options:

--help	Show help	[boolean]
--version	Show version number	[boolean]
--title	Başlık1	[string] [required]
--body	gövde	[string] [required]

```
PS C:\Users\abird\OneDrive\Masaüstü\Node_JS_lab\Hafta 2>
```

```
PS C:\Users\abird\OneDrive\Masaüstü\Node_JS_lab\Hafta 2> node h2u3.js ekle --title="başlık" --body="gövde"
başlık
gövde
{ _: [ 'ekle' ], title: 'başlık', body: 'gövde', '$0': 'h2u3.js' }
```

# .json dosyası oluşturma

- Burada önce bir JavaScript nesnesi (book) tanımlanıyor. Sonra bu nesne JSON formatına çevriliyor (JSON.stringify(book)) ve bir string olarak saklanıyor. Daha sonra JSON stringi tekrar nesneye (JSON.parse(bookJSON)) dönüştürülüyor.

```
PS C:\Users\abird\OneDrive\Masaüstü\Node_JS_lab\Hafta 2> node .\h2u4.js
[object Object]

1924

{"title":"başlık","author":"gövde","year":1924}

[object Object]

undefined
```

```
JS h2u4.js > [bookJSON]
1  /* JavaScript nesnesi
2  ✓ const book = {
3      title : 'başlık',
4      author: 'gövde',
5      year: 1924
6  }
7
8  console.log(book + "\n")
9  console.log(book.year + "\n")
10
11  /* JSON
12  const bookJSON = JSON.stringify(book)
13  console.log(bookJSON + "\n")
14
15  /* JSON Parse func
16  const parsedBook = JSON.parse(bookJSON)
17  console.log(parsedBook + "\n")
18
19  /* JSON İçeriğine erişme
20  console.log(bookJSON.author + "\n") //! tanımlanamaz hatası
```

## ► Başka bir örnek

```
JS h2u5.js > ...
1  const fs = require("fs")
2  const chalk = require("chalk")
3
4  const book = {
5    title : 'Harry Potter and The Chamber of the Secrets',
6    author: 'J. K. Rowling',
7    year: 1998
8  }
9
10 const bookJSON = JSON.stringify(book)
11
12 /* write json file
13 fs.writeFileSync('book.json',bookJSON)
14
15 /* read a buffer
16 const dataBuffer = fs.readFileSync('book.json')
17 console.log(chalk.red('Data Buffer:'))
18 console.log(dataBuffer)
19
20 const data = dataBuffer.toString()
21 console.log(chalk.red('Data:'))
22 console.log(data)
23 const parsedData = JSON.parse(data)
24 console.log(chalk.red('Parsed Data:'))
25 console.log(parsedData)
```

```
PS C:\Users\abird\OneDrive\Masaüstü\Node_JS_lab\Hafta 2> node .\h2u5.js
Data Buffer:
<Buffer 7b 22 74 69 74 6c 65 22 3a 22 48 61 72 72 79 20 50 6f 74 74 65 72 20 61 6e 64 20 54 68 65 20 43 68 61 6d 62 65 72 20 6f 66 20 74 68 65 20 53 65 63 72 ... 42 more bytes>
Data:
{"title":"Harry Potter and The Chamber of the Secrets","author":"J. K. Rowling","year":1998}
Parsed Data:
{
  title: 'Harry Potter and The Chamber of the Secrets',
  author: 'J. K. Rowling',
  year: 1998
}
```



► Son