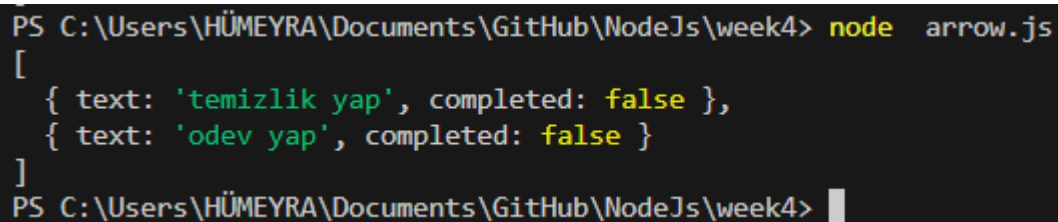


NODE-JS 3

1. adım :

arrow_test.js adlı dosyayı oluşturalım :

```
const tasks = {
  tasks: [{
    text: 'alışveriş yap',
    completed: true
  }, {
    text: 'temizlik yap',
    completed: false
  }, {
    text: 'ödev yap',
    completed: false
  }],
  getTasksToDo() {
    const tasksToDo = this.tasks.filter((task) => {
      return task.completed === false
    })
    return tasksToDo
  }
}
console.log(tasks.getTasksToDo())
```



```
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week4> node arrow.js
[
  { text: 'temizlik yap', completed: false },
  { text: 'ödev yap', completed: false }
]
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week4>
```

tamalanmamış görevler döndürülür.

```
const tasks = {
  tasks: [{
    text: 'alışveriş yap',
    completed: true
  }, {
    text: 'temizlik yap',
    completed: false
  }, {
    text: 'ödev yap',
    completed: false
  }],
  getTasksToDo() {
    return this.tasks.filter((task) => {
      return task.completed === false
    })
  }
}
console.log(tasks.getTasksToDo())
```

```

const tasks = {
  tasks: [
    { text: 'Alışveriş yap', completed: true },
    { text: 'Temizlik yap', completed: false },
    { text: 'Ödev yap', completed: false }
  ],
  getTasksToDo() {
    return this.tasks.filter(task => !task.completed).map(task =>
task.text);
  },
  getTasksToDoTR() {
    return this.tasks.filter(task => task.completed===false);
  }
};

console.log(tasks.getTasksToDo()); // Alışveriş yap, Temizlik yap, Ödev
yap
console.log(tasks.getTasksToDoTR()); // Alışveriş yap, Temizlik yap,
Ödev yap

```

```

PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week4> node arrow.js
[ 'Temizlik yap', 'Ödev yap' ]
[
  { text: 'Temizlik yap', completed: false },
  { text: 'Ödev yap', completed: false }
]
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week4> node arrow.js
[ 'Temizlik yap', 'Ödev yap' ]
[
  { text: 'Temizlik yap', completed: false },
  { text: 'Ödev yap', completed: false }
]
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week4>

```

```

NodeJs > week2 > JS arrow_test.js > ...
Click here to ask Blackbox to help you code faster
1  const tasks = {
2    tasks: [
3      { text: 'Alışveriş yap', completed: true },
4      { text: 'Temizlik yap', completed: false },
5      { text: 'Ödev yap', completed: false }
6    ],
7    getTasksToDo() {
8      return this.tasks.filter(task => !task.completed).map(task => task.text);
9    },
10   getTasksToDoTR() {
11     return this.tasks.filter(task => task.completed===false);
12   }
13
14 };
15
16 console.log(tasks.getTasksToDo()); // Alışveriş yap, Temizlik yap, Ödev yap
17 console.log(tasks.getTasksToDoTR()); // Alışveriş yap, Temizlik yap, Ödev yap
18

```

2.adım

Kodlar için iyileştirmeler yapıldı note.js dosyasındaki fonksiyonlar dönüştürldü.

```
// Eski hali
handler: function (argv) {
  // kodlar
}

// Yeni hali
handler(argv) {
  // kodlar
}
```

add komutunun handlerının düzenlenmesi:

```
// komut girildiğinde ne yapılacağını belirlenmesi
handler (argv) {
  notes.addNote(argv.title, argv.body);
  // console.log("başlık:", argv.title);
  // console.log("title içeriği:", argv.body);
}
```

addNote fonksiyonu dönüşüm işlemi:

```
// const addNote = function(title, body) { conver to arrow fonk
const addNote = (title, body) =>{
  const notes = loadNotes(); // bir dizi döndürecektir
86 | const listNotes=()=>{
87 |   const notes = loadNotes();
88 |   console.log(chalk.yellow("\nYour Notes: \n\n"))
89 |   notes.forEach((note) => console.log(`${chalk.blue(note.title)} : ${chalk.white(note.body)}`));
90 | }
91 | module.exports = {
92 |   addNote: addNote,
93 |   getNotes: getNotes,
94 |   removeNote: removeNote,
95 |   listNotes: listNotes
96 | };
```

3.adım

addNote fonksiyonunda iyileştirmeler yapıldı

```
const duplicateNote = notes.find((note) => note.title === title);  
  
// const duplicatesNotes = notes.filter( (note) => note.title === title);
```

filter fonksiyonu mevcut tüm notların parametre olarak verilen title ile eşleşmesini kontrol eder. (ilk not title eşleşmiş olsa bile tüm not lar kontrol yapar)

find ise title eşleşmesini bulduğunda durur tüm notları kontrol etmez

```
const notes = loadNotes(); // bu dizi döndürecek  
// const duplicatesNotes = notes.filter( (note) => note.title === title);  
  
//performans artımı için kayıtlı her not filter ile bakılıyor tüm elemanlara bakmasın bulunca bakmayı bıraksın !  
const duplicatesNotes=notes.find((note) => note.title === title );  
  
console.log(duplicatesNotes);  
// aynı title bir note varsa onu bulur döndürüyorsa undefined olur  
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2> node app2.js add --title="hafta 4 ödevini yap " --body="pazar gunune tamamlama "  
undefined  
note added!  
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2> node app2.js add --title="hafta 4 ödevini yap " --body="pazar gunune tamamlama "  
{ title: 'hafta 4 ödevini yap ', body: 'pazar gunune tamamlama ' }  
note already added !  
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2> |
```

```
// const addNote = function(title, body) { conver to arrow fonk  
const addNote = (title, body) =>{  
    const notes = loadNotes(); // bir dizi döndürecek  
    // const duplicatesNotes = notes.filter( (note) => note.title === title);  
  
    //performans artımı için kayıtlı her not filter ile bakılıyor tüm elemanlara bakmasın  
    bulunca bakmayı bıraksın !  
    const duplicatesNotes=notes.find((note) => note.title === title );  
  
    // notes'ların özgün olmasını nasıl sağlanır?  
    // notes içindeki her eleman için tekrar tekrar çalıştırılır  
    // const duplicatesNotes = notes.filter(function (note) {  
    //     return note.title === title;  
    // });  
    // ARROW FONK AD DÖNÜŞTÜRÜLDÜ  
  
    // if (duplicatesNotes.length === 0) {  
    //     // array sonuna obje ekler  
  
    console.log(duplicatesNotes); // aynı title bir note varsa onu bulur döndürüyorsa  
    undefined olur  
  
    debugger  
    if(!duplicatesNotes){ // tekrarlı not yoksa aynı title yoksa eklenecek  
        // if(duplicatesNotes===undefined)  
        notes.push({  
            // notes arasındaki title  
            title: title, // fonksiyondaki title  
            body: body  
        });  
        // console.log(notes)  
        saveNotes(notes);  
        console.log(chalk.greenBright.inverse("note added! "));  
    } else {  
        console.log(chalk.redBright.inverse("note already added ! "));  
    }  
};
```

4. adım

list ve read komutları için işlevsellik sağlandı.

listNote

notes.js içine listNote fonksiyonu yazıldı ve exportedildi.

app2.js içinde handler olarak güncellendi

```
yargs.command({
  command: 'list',
  describe: "notları listeler",
  handler() {
    notes.listNotes();
  }
})

99 const listNotes={()=>{
100   const notes = loadNotes();
101
102   console.log(chalk.inverse.yellow("\n kayıtlı notlar: \n\n"))
103   // arrayin her elemanına note olarak ele alındı .
104   notes.forEach((note) => console.log(`${chalk.blue(note.title)} : ${chalk.white(note.body)}`));
105 }
```

```
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2> node app2.js list

kayıtlı notlar:

aaa : bb
hellllooo : hello2222
ben : hello2222
sssben : hello2222
ssssben : hello2222
hafta 4 ödevini yap : pazar gunune tamamla
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2>
```

readNotes

```
64 yargs.command({
65   command: 'read',
66   describe: "seçilen notu gösterir",
67   // handler: function () {
68   //   console.log("Seçili not okunuyor..");
69   // }
70   // ARROW FONK A DÖNÜŞTÜRÜLDÜ
71   builder: {
72     title: {
73       describe: 'not basligi silinir.',
74       demandOption: true,
75       type: "string"
76     }
77   },
78   handler (argv) {
79     notes.readNotes(argv.title);
80   }
81 },
82 // ARROW FONK A DÖNÜŞTÜRÜLDÜ
83 builder: {
84   title: {
85     describe: 'not okunur.',
86     demandOption: true,
87     type: "string"
88   }
89 },
90 handler (argv) {
91   notes.readNotes(argv.title);
92 }
```

```
const readNotes=(title)=>{
  // const notes = loadNotes().find( note=>note.title===process.argv[2]);
  const note = loadNotes().find( note=>note.title===title);
  if(note){
    console.log(chalk.inverse.greenBright(note.title));
  }else{
    console.log(chalk.inverse.redBright( "geçersiz istek" ));
  }
}
```

!!! Yazılan fonksiyonlar export edildi.

```
module.exports = {
  addNote: addNote,
  // getNotes: getNotes,
  removeNote: removeNote,
  listNotes: listNotes,
  readNotes: readNotes
};
```

debugger

```
JS notes.js M X JS app2.js M JS arrow.js U
NodeJs > week2 > JS notes.js > [0] addNote
5  const addNote = (title, body) =>{
18
19  // if (duplicatesNotes.length === 0) {
20  // array sonuna obje ekler
21
22  console.log(duplicatesNotes); // aynı title bir note v
23
24  debugger
```

```
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs> node --inspect-brk app2.js add --title="Course" --body="nodejs"
Debugger listening on ws://127.0.0.1:9229/c1c6c259-d09d-4bb8-89b3-8b84abf302f6
For help, see: https://nodejs.org/en/docs/inspector
```

Artık hata ayıklama işlemine başlayabiliriz. Tarayıcınızı açarak `chrome://inspect/` adresine gidin ve "Remote target" altında işlemi seçin. "Sources" sekmesine geçerek `app2.js` dosyasını görebilirsiniz.

Devices

☒ Discover USB devices

Port forwarding...

☒ Discover network targets

Configure...

[Open dedicated DevTools for Node](#)

Remote Target #LOCALHOST

ts

Target (v20.11.1) [trace](#)

`app2.js` [file:///C:/_Users_H%C3%9CMEYRA_Documents_GitHub_NodeJs_week2_app2.js](#)
[inspect](#)

kayıtlı notlar:

[VM152 notes.js:102](#)

aaa : bb	VM152 notes.js:104
hellllooo : hello2222	VM152 notes.js:104
ben : hello2222	VM152 notes.js:104
sssbenn : hello2222	VM152 notes.js:104
sssssbenn : hello2222	VM152 notes.js:104
hafta 4 ödevini yap : pazar gunune tamamla	VM152 notes.js:104
Course : nodejs	VM152 notes.js:104

```
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs> cd week2
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2> node --inspect-brk app2.js list
Debugger listening on ws://127.0.0.1:9229/3840ec30-6f4f-49c3-8489-f6047ed38b19
For help, see: https://nodejs.org/en/docs/inspector
Debugger attached.
```

kayıtlı notlar:

```
aaa : bb
hellllooo : hello2222
ben : hello2222
sssbenn : hello2222
sssssbenn : hello2222
hafta 4 ödevini yap : pazar gunune tamamla
Course : nodejs
Waiting for the debugger to disconnect...
```


Test

```
note already added !
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2> node inspect app2.js add --title="ssssben" --body="hello2222"
< Debugger listening on ws://127.0.0.1:9229/515f085c-bfeb-4328-8997-0f181cdd9c26
< For help, see: https://nodejs.org/en/docs/inspector
<
connecting to 127.0.0.1:9229 ... ok
< Debugger attached.
<
Break on start in app2.js:2
  1 // import yargs module
> 2 const command = require("nodemon/lib/config/command");
  3 const yargs= require("yargs");
  4
debug>

PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2> node app2.js add --title="hafta4" --body="odev gönder"
undefined
note added!
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2> node app2.js add --title="hafta4" --body="odev gönder"
{ title: 'hafta4', body: 'odev gönder' }
note already added !
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2> node app2.js list

kayıtlı notlar:

aaa : bb
hellllooo : hello2222
ben : hello2222
sssbens : hello2222
ssssben : hello2222
hafta 4 ödevini yap : pazar gunune tamamla
hafta4 : odev gönder
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2> node app2.js read

PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2> node app2.js read --title="hafta4"
hafta4
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2> node app2.js remove --title="hafta4" --body="odev gönder"
Note removed!
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2> node app2.js remove --title="hafta4" --body="odev gönder"
Note not found!
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2> node app2.js list

kayıtlı notlar:

aaa : bb
hellllooo : hello2222
ben : hello2222
sssbens : hello2222
ssssben : hello2222
hafta 4 ödevini yap : pazar gunune tamamla
PS C:\Users\HÜMEYRA\Documents\GitHub\NodeJs\week2>
```

BİLGİSAYAR MÜHENDİSİ HÜMEYRA ÇİMEN

Uygulama dosyalarına ilgili bağlantıdan ulaşabilirsiniz:

<https://github.com/hmyrcmn/NodeJs/tree/main/week2>