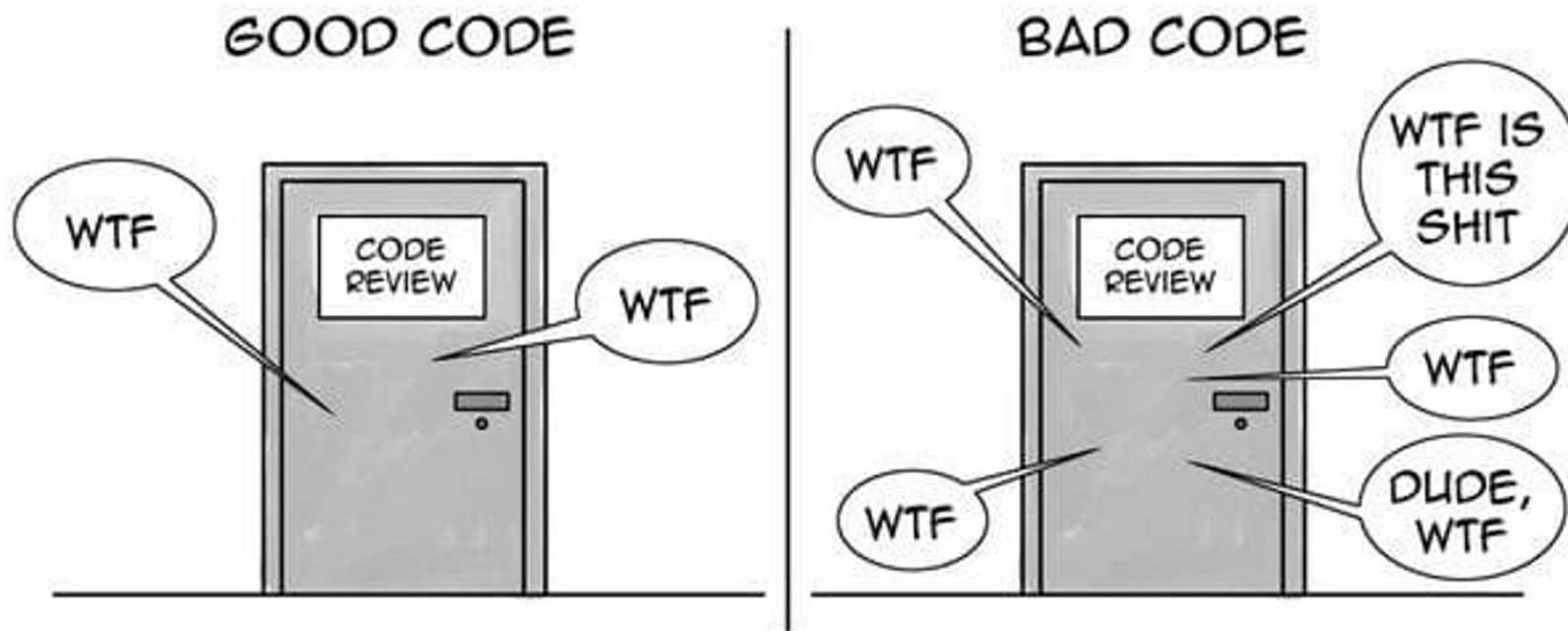


Kaip nustatyti ar kodas geras?



Naming

Atskleiskite intenciją

✗

```
const s; // elapsed time in seconds
```




✓

```
const elapsedTimeInSeconds;  
const daysSinceCreation;  
const hoursSinceModification;  
const fileAgeInDays;
```

Naming

Venkite desinformacijos, nenaudokite tokių simbolių, kurie gali klaidinti.



```
const a = 1;

if (0 == 1) {
    a = 01;
} else {
    1 = 01;
}
```



Naming

Make meaningful distinctions

✖

```
function copyNumbers(a1, a2) {  
  for (let i = 0; i < a1.length; i++){  
    a2[i] = a1[i];  
  }  
}
```



✔

```
function copyNumbers(source, destination) {  
  for (let i = 0; i < source.length; i++){  
    destination[i] = source[i];  
  }  
}
```

Naming

Make meaningful distinctions

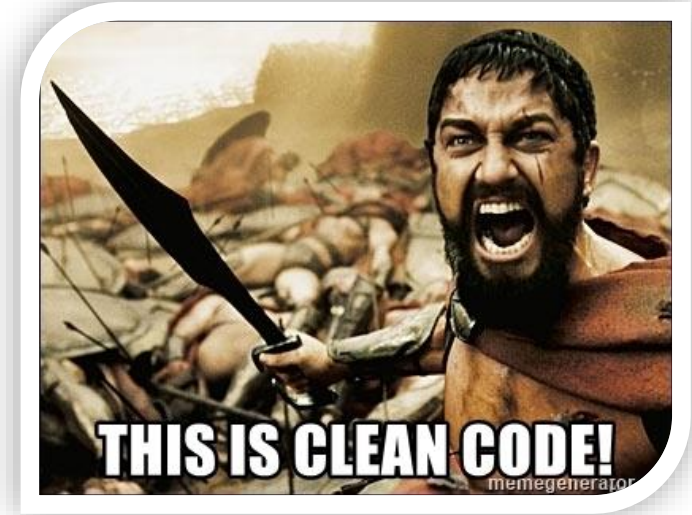
- ◆ moneyAmount is indistinguishable from money,
- ◆ customerInfo is indistinguishable from customer,
- ◆ accountData is indistinguishable from account,
- ◆ theMessage is indistinguishable from message,
- ◆ accountList is indistinguishable from accounts



Naming

Naudokite randamus pavadinimus

- ◆ Projekte lengva surasti `MIN_TASKS_PER_STUDENT`, bet sunku rasti skaičių 8
- ◆ Kintamojo pavadinimas e yra blogas pasirinkimas vien dėl to, kad jį surasti yra be galo sunku. Taip pat tai yra labiausiai naudojama raidė anglų kalboje.
- ◆ Ilgesni pavadinimai > Trumpesni pavadinimai



Naming

Vienas pavadinimas per projektą

- ◆ Painu turėti fetch, retrieve ir get metodus viename projekte skirtinguose failuose.
- ◆ Taip pat ir su controller, manager, driver.



Naming

Good

DeleteItems()

Bad

Kaboom()

Good

kill()

Bad

holyHandGranade()

Good

abort()

Bad

eatMyShorts()

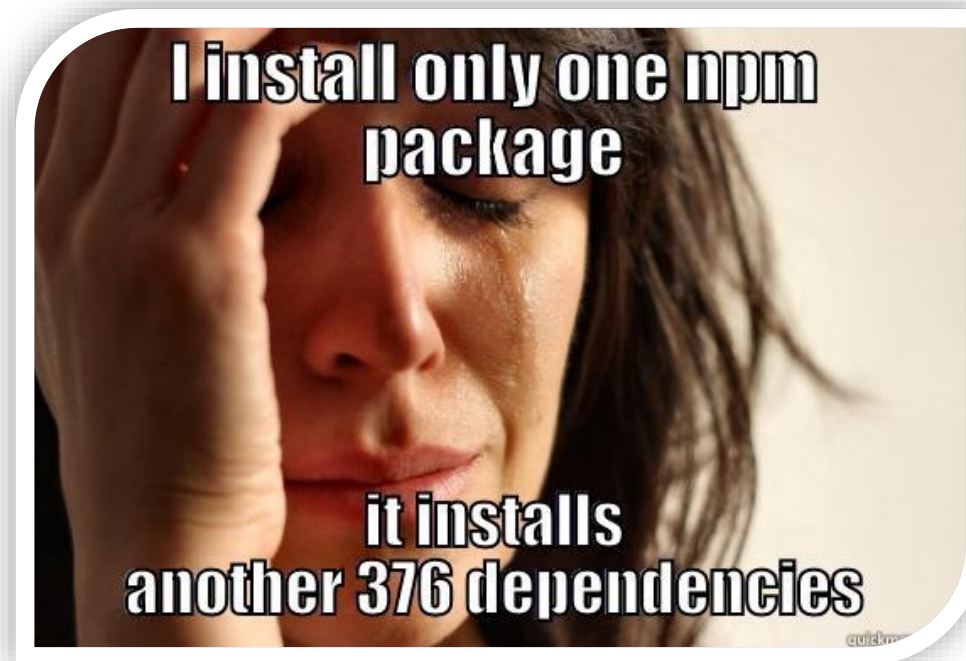


FRONT-END
Academy
by Present Connection



npm

- ◆ Node.js paketas susideda iš visų failų kurių reikia moduliui
- ◆ Npm yra paketų tvarkyklė skirta Node.js paketams
- ◆ www.npmjs.com - puslapis kuriame yra tūkstančiai nemokamų paketų
- ◆ NPM sukuria aplanką „node_modules“, kuriame yra visi paketai
- ◆ **node_modules** nekeliauja į source code



Npm komandos

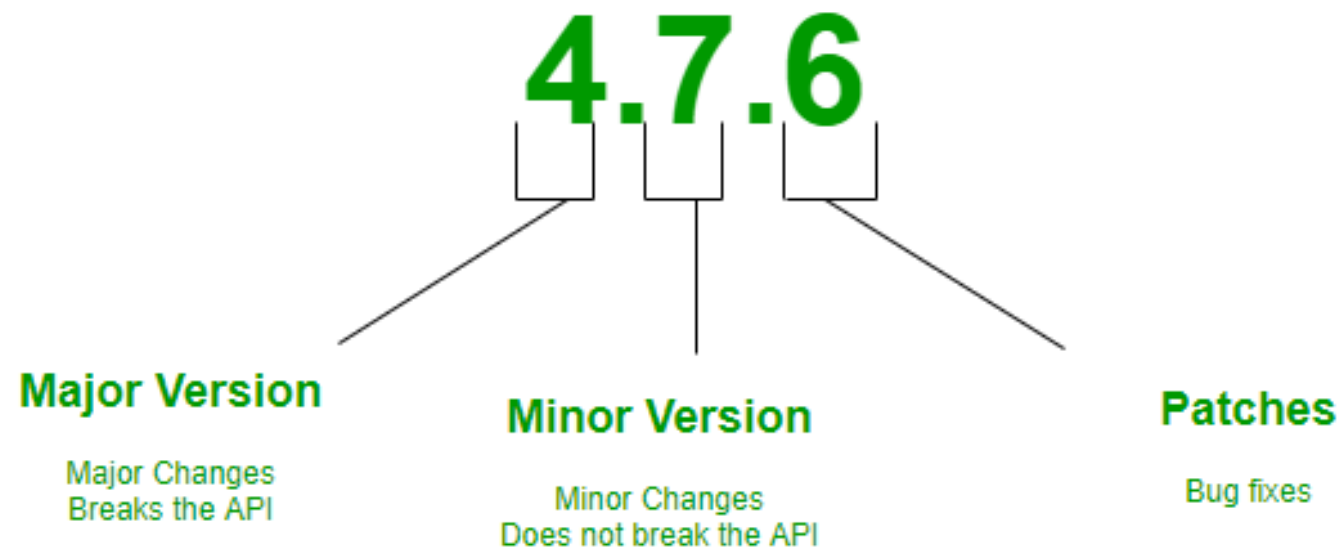
Komanda	Aprašymas
npm	
init	Sukuria package.json failą
install	Diegia paketą
uninstall	Pašalinti paketą
build	Sukuria paketą
help	Gauti pagalbą
run	Paleisti skriptą
update	Atnaujinti paketą
start	Paleisti paketą
stop	Sustabdyti paketą

package.json

- ◆ Metadata apie aplikacijos paketus
- ◆ Nusako paketų dependencies

```
{
  "name": "academy-app",
  "version": "1.0.0",
  "description": "First front-end academy in PC",
  "repository": {
    "type": "git",
    "url": "https://github.com/PcFrontEndAcademy"
  },
  "engines": {
    "npm": ">=3",
    "node": ">=5"
  },
  "author": "Ernestas Burokas",
  "license": "MIT",
  "scripts": {},
  "dependencies": {
    "express": "^4.16.2",
    "jquery": "^3.2.1"
  }
}
```

Modulių versijavimas



Modulių versijavimas

```
"dependencies": {  
  "tasklist": "3.3.1"  
}
```

- 2.1.0
- 2.2.0
- 2.2.1
- 2.3.0
- 2.4.0
- 2.4.1
- 3.0.0
- 3.0.1
- 3.1.0
- 3.2.0
- 3.3.0
- 3.3.1

- 3.4.0
- 3.5.0
- 3.6.0
- 1.0.2
- 3.7.0
- 2.4.2
- 3.8.0
- 3.9.0
- 3.9.1
- 3.9.2
- 3.9.3
- 3.10.0

- 3.10.1
- 4.0.0
- 4.0.1
- 4.1.0
- 4.2.0
- 4.2.1
- 4.3.0
- 4.4.0
- 4.5.0
- 4.5.1
- 4.6.0
- 4.6.1

Modulių versijavimas [^]

```
"dependencies": {  
  "tasklist": "^3.3.0"  
}
```

- 2.1.0
- 2.2.0
- 2.2.1
- 2.3.0
- 2.4.0
- 2.4.1
- 3.0.0
- 3.0.1
- 3.1.0
- 3.2.0
- 3.3.0
- 3.3.1

- 3.4.0
- 3.5.0
- 3.6.0
- 1.0.2
- 3.7.0
- 2.4.2
- 3.8.0
- 3.9.0
- 3.9.1
- 3.9.2
- 3.9.3
- 3.10.0

- 3.10.1
- 4.0.0
- 4.0.1
- 4.1.0
- 4.2.0
- 4.2.1
- 4.3.0
- 4.4.0
- 4.5.0
- 4.5.1
- 4.6.0
- 4.6.1

Modulių versijavimas [~]

```
"dependencies": {  
  "tasklist": "~3.3.0"  
}
```

- | | | |
|---------|----------|----------|
| • 2.1.0 | • 3.4.0 | • 3.10.1 |
| • 2.2.0 | • 3.5.0 | • 4.0.0 |
| • 2.2.1 | • 3.6.0 | • 4.0.1 |
| • 2.3.0 | • 1.0.2 | • 4.1.0 |
| • 2.4.0 | • 3.7.0 | • 4.2.0 |
| • 2.4.1 | • 2.4.2 | • 4.2.1 |
| • 3.0.0 | • 3.8.0 | • 4.3.0 |
| • 3.0.1 | • 3.9.0 | • 4.4.0 |
| • 3.1.0 | • 3.9.1 | • 4.5.0 |
| • 3.2.0 | • 3.9.2 | • 4.5.1 |
| • 3.3.0 | • 3.9.3 | • 4.6.0 |
| • 3.3.1 | • 3.10.0 | • 4.6.1 |

Modulių versijavimas [>]

```
"dependencies": {  
  "tasklist": ">3.3.0"  
}
```

- 2.1.0
- 2.2.0
- 2.2.1
- 2.3.0
- 2.4.0
- 2.4.1
- 3.0.0
- 3.0.1
- 3.1.0
- 3.2.0
- 3.3.0
- 3.3.1

- 3.4.0
- 3.5.0
- 3.6.0
- 1.0.2
- 3.7.0
- 2.4.2
- 3.8.0
- 3.9.0
- 3.9.1
- 3.9.2
- 3.9.3
- 3.10.0

- 3.10.1
- 4.0.0
- 4.0.1
- 4.1.0
- 4.2.0
- 4.2.1
- 4.3.0
- 4.4.0
- 4.5.0
- 4.5.1
- 4.6.0
- 4.6.1

Modulių versijavimas [-]

```
"dependencies": {  
  "tasklist": "3.3.0 - 3.8.0"  
}
```

- 2.1.0

- 2.2.0

- 2.2.1

- 2.3.0

- 2.4.0

- 2.4.1

- 3.0.0

- 3.0.1

- 3.1.0

- 3.2.0

- 3.3.0

- 3.3.1

- 3.4.0

- 3.5.0

- 3.6.0

- 1.0.2

- 3.7.0

- 2.4.2

- 3.8.0

- 3.9.0

- 3.9.1

- 3.9.2

- 3.9.3

- 3.10.0

- 3.10.1

- 4.0.0

- 4.0.1

- 4.1.0

- 4.2.0

- 4.2.1

- 4.3.0

- 4.4.0

- 4.5.0

- 4.5.1

- 4.6.0

- 4.6.1

package-lock.json

- ◆ Atsirado nuo 5 npm versijos
- ◆ Apibūdina tikslų projekto paketų medį taip vėlesnes instaliacijas paversdamas identiškomis.
- ◆ Generuojasi automatiškai įrašant paketus
- ◆ Tiek package.json tiek package-lock.json turi būti source control'e



Dev dependencies

- ◆ Moduliai kurie naudojami tik development'ui
- ◆ Tokie moduliai nebus įtraukti kai bus atliekamas jų instaliavimas production'ui
- ◆ Dependencies: jquery, express, react
- ◆ Dev Dependencies: babel, eslint
- ◆ `npm i <package_name> --save-dev`



Dev dependencies

```
devDependencies: {  
  "add-asset-html-webpack-plugin": "2.0.1",  
  "babel-cli": "^6.26.0",  
  "coveralls": "2.13.1",  
  "css-loader": "0.28.4",  
  "enzyme": "2.8.2",  
  "eslint": "3.19.0",  
  "style-loader": "0.18.1",  
  "url-loader": "0.5.8",  
  "webpack": "3.5.5",  
  "webpack-dev-middleware": "1.11.0",  
  "webpack-hot-middleware": "2.18.0"  
}
```

Globalūs moduliai

- ◆ Lokālūs moduliai yra įrašomi projekto scope (./node_modules), tad juos galima naudoti tik projekte.
- ◆ Globalūs moduliai yra įrašomi globaliai (%AppData%/npm) ir juos galima naudoti visuose projektuose.
- ◆ -g
- ◆ Naudojami dėl to kad lokālūs moduliai nėra pasiekiam CLI.



Globalūs moduliai [trūkumai]

- ◆ Neatsiras package.json faile
- ◆ Sunku sugaudyti modulio versijas
- ◆ Venkime globalių modulių!
- ◆ Norint paleisti lokalių modulių terminale, pasirašome tam skriptą.

```
"scripts": {  
  "your-package": "your-package",  
  "build": "babel es6 -d es5"  
}
```



npx

- ◆ CLI įrankis, kuris padeda lengviau valdyti modulių, kurie yra sutalpinti npm.
- ◆ Nereikia skripto, kad paleisti lokalų modulį, užtenka **npx you package**
- ◆ Galima paleisti modulį, kuris nebuvo įrašytas
- ◆ Skirtingų modulių ar jų versijų testavimas



DEMO



Workshop

- ◆ Firebase-> registruotis ir susikurti naują projektą
- ◆ Firebase -> Sukurti firestore ir app
- ◆ npm i firebase
- ◆ Sukurti naują studentą iš kodo
- ◆ Gauti visus studentus



Namų darbai

- ◆ Pradėti savo projekto back end (C#, Node.js, Firebase) struktūrą dėti. Į struktūrą įeina DB, CRUD.

