

# JavaScript ohjelmointi

*Markdown ja Git*

MARGIT TENNOSAARIN MATERIAALISTA  
LAURA JÄRVISEN MUOKKAAMA

# Markdown

Markdown on kevyt merkintäkieli, jonka avulla voi luoda muotoiltua tekstiä käyttämällä pelkän tekstin editoria.

Markdown-tiedostojen tunnisteet ovat **.md** ja **.markdown**.

Markdown-kieltä käytetään usein README-tiedostojen luomiseen.

Alla kaksi linkkiä, joista voit katsoa Markdownin syntaksia eli tapaa muotoilla asiat:

<https://docs.github.com>

[https://github.com/margittennosaar/markdown\\_study\\_materials](https://github.com/margittennosaar/markdown_study_materials)

# .md syntaksi esimerkkejä

```
# Isoin otsikko * kohta 1
## Pienempi otsikko * kohta 2
*Tämä teksti tulee kursiivilla* * kohta 2a
**Tämä teksti tulee lihavoituna** * kohta 2b
_On **mahdollista** yhdistää niitä_ 1.kohta 1
~~Tämä teksti viivataan yli~~ 2. kohta 2
                                         3. kohta 3
                                         - kohta 3a
                                         - kohta 3b
```

Kokeile itse!

Kokeile itse!

# Syntaksi jatkuu

![Markdown Logo](Markdown-mark.svg)  
Format: ![Alt Text](url)

<http://github.com>  
[GitHub](http://github.com)

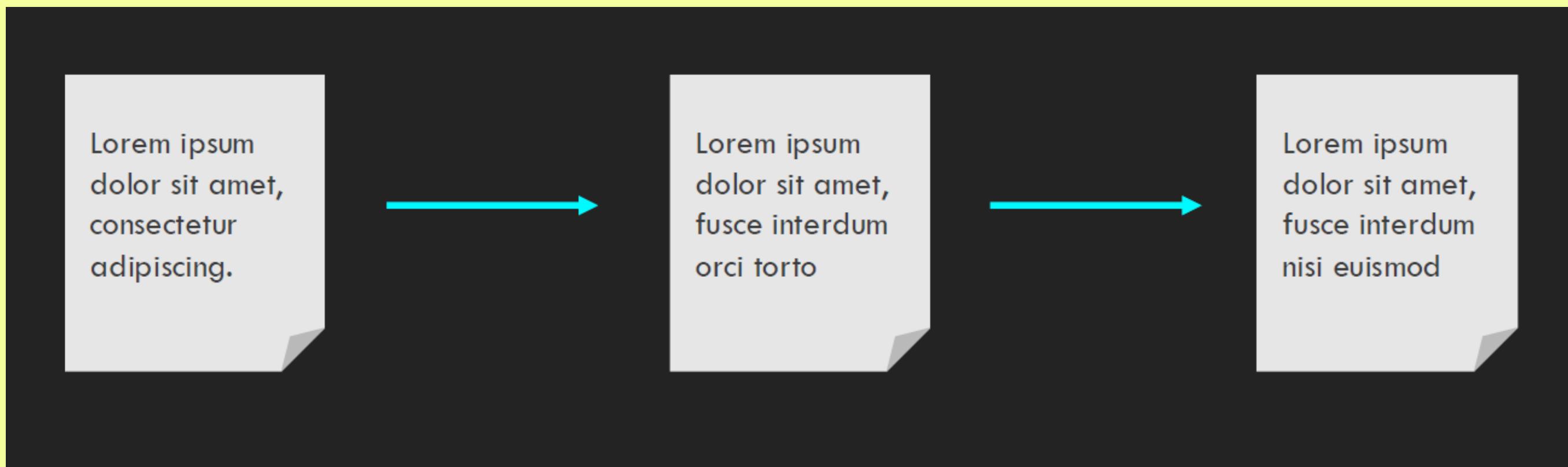
```
```js
function fancyAlert(arg) {
if(arg) {
$.facebox({div:'#foo'})
}
}```
```

# *Markdown tehtävä*

- Luo uusi tiedosto nimeltä README.md
- Kirjoita tiedostoon, mitä tämä projekti sisältää (kekси itse)
- Käytä erilaisia Markdown-syntakseja harjoitellaksesi

# *Versionhallinta*

Versionhallinta viittaa menetelmiin, joilla tiedostot säilytetään muistissa ja mitä ne ovat olleet missäkin vaiheessa, kuka on niitä muokannut, milloin ja miten (ja usein vapaamuotoisia kuvaauksia siitä, miksi).



# Välijulkomuksien Omat nimet

*Repository, kansio ja hakemisto*  
tarkoittavat samaa asiaa.

Välillä - varsinkin  
versionhallinnassa - kansioon  
voidaan viitata myös *tiedostona*.

*Työkalut*

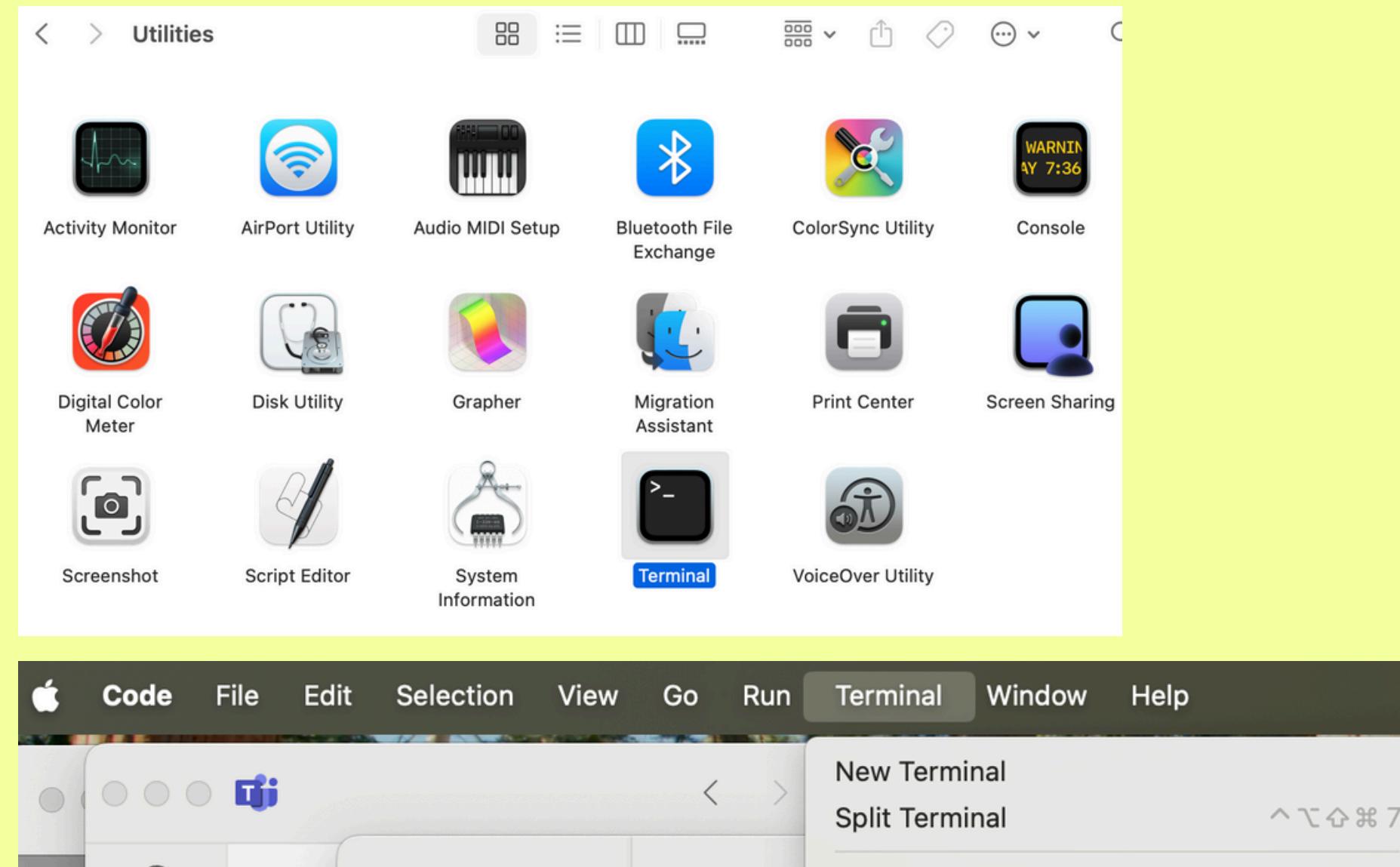
Git

CLI

GitHub

# Välijulkomuus

*Terminaalia eli komentoriviä* voi käyttää monesta eri paikkaa.  
Maceissa on oma Terminal-apuohjelma,  
VSCoden sisällä on myös terminaali.



# *Komentorivillä liikkuminen*

**pwd**

*Print Working Directory* - Näyttää polun nykyiseen sijaintiin

**ls**

Listaa kansiot ja tiedostot nykyisessä sijainnissa.

**cd [kansio]**

*Change Directory* - siirtyy määriteltyyn kansioon.

**cd ..**

Siirtyy ylempään kansioon.

# *Resurssien hallinta komentorivillä*

**mkdir [kansion\_nimi]**

*Make Directory* - Luo uuden hakemiston

**touch [tiedoston\_nimi]**

Luo uuden tiedoston, jos sellaista ei vielä ole

**rm [tiedoston\_nimi]**

Poistaa tiedoston.

**rm -r [kansion\_nimi]**

Poistaa kansion ja sen koko sisällön (rekursiivisesti)

**cp [lähde] [kohde]**

Kopioi tiedoston tai kansion

**mv [lähde] [kohde]**

Siirtää tiedoston/kansion tai muuttaa tiedoston/kansion nimeä

# *Git komentoriviltä*

## **git init**

Alustaa uuden git-repositorion

## **git clone [url]**

Kloonaa repositorion uuteen hakemistoon

## **git status**

Näyttää muutosten tilan: seuraamattomat, muokatut tai valmistellut

## **git add [tiedosto/kansio]**

Lisää tiedoston tai hakemiston valmistelualueelle

## **git commit -m "[commit viesti]"**

Tallentaa tiedostojen tilannekuvan versionhallinnan historiaan

## **git pull**

Noutaa muutokset etäpalvelimelta ja yhdistää ne nykyiseen haaraan

## **git push**

Lataa kaikki paikallisen haaran commitit etärepositioon

# *Git tehtävä*

1. Luo GitHub tunnus
2. Asenna Git
3. Konfiguroi Git

```
git config --global user.name "Oma Nimesi"
```

```
git config --global user.email "oma.osoitteesi@example.com"
```