

Päivä 3: Erilaisia ohjelmointikieliä

CODE BOOTCAMP KYMENLAAKSO 2020
JANI JÄRVINEN, INTERTECHNO TRAINING OY



Ohjelmistoasennuksia

- Päivän aihepiiriin kuuluu erilaisten ohjelmointikielten – erityisesti JavaScriptiin – tutustumista
- Voidaksemme käyttää JavaScriptiä, tarvitsemme Node.js-nimisen ympäristön
- Asennamme myös Python-ympäristön, jotta voimme testata Python-ympäristön käyttöä
- Asennuslinkit
 - Node.js: <https://nodejs.org/en/>
 - Python: <https://www.python.org/downloads/>

Erilaisia ohjelmointikieliä

Erilaisia ohjelmointikieliä on paljon

- Erilaisia ohjelmointikieliä on tuhansittain, mutta koko ohjelmointiuransa voi pärjätä jopa yhdellä kielellä, jos valitsee viisaasti
- Tyypillistä kuitenkin on, että kehittäjä osaa muutamia (3–6) kieliä riittävän hyvin
- Eri tyyppisiä ohjelmointimalleja eli paradigmoja
 - Imperatiivinen ohjelmointi
 - Olio-ohjelmointi
 - Funktionaalinen ohjelmointi
- Yleisimpiä ohjelmointikieliä
 - <https://www.tiobe.com/tiobe-index/>
 - <https://spectrum.ieee.org/static/interactive-the-top-programming-languages-2018>

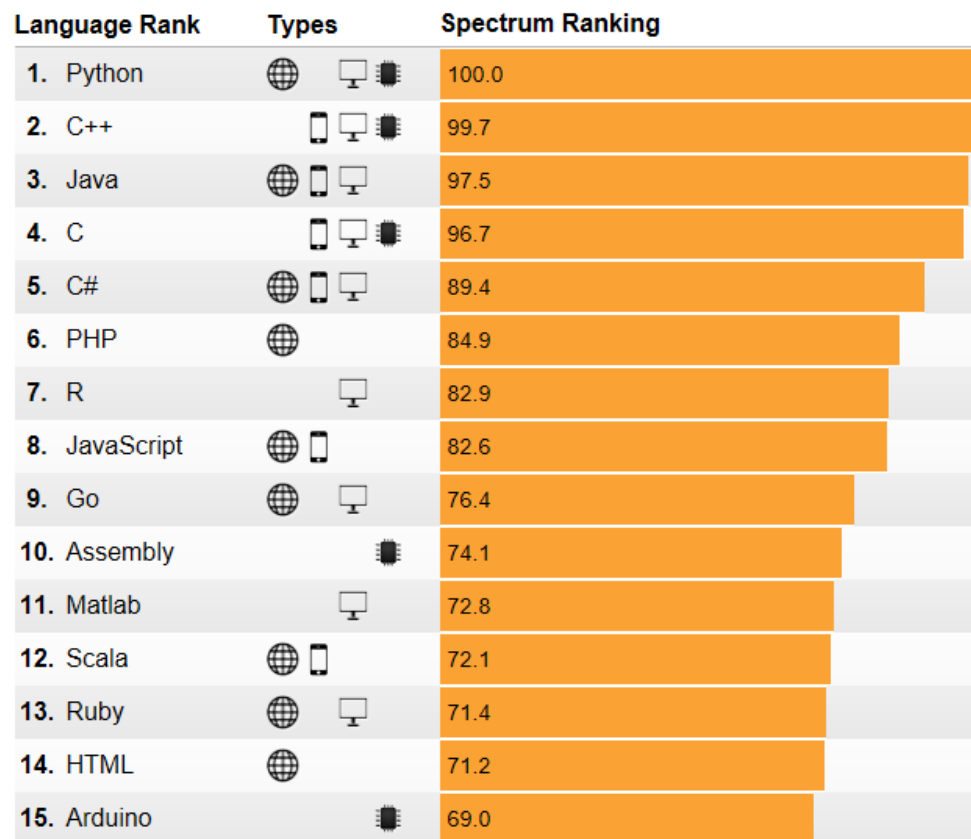
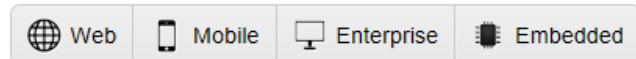
Mitä kieliä kannattaisi opetella?

- Jokin C-pohjainen kieli on aina hyvä lähtökohta
 - Esimerkiksi C, C++, Java tai C#
- JavaScript tai jokin sen muunnelma, kuten TypeScript
- Jokin tietokantoihin liittyvä kieli, yleensä vähintään SQL-kieli
- Web-ohjelmoinnin peruskielet: HTML sekä CSS
- Jokin skriptikieli, kuten Python, PowerShell tai AWK
- Muista: kun osaat hyvin yhden ohjelmointikielen, on helpompaa opetella toinen, kolmas, ...

Koodin suorittaminen

- Lähtökohta on, että ohjelmakoodia suoritetaan vasemmalta oikealle, ylhäältä alas
- Joissakin ohjelmointikielissä isoilla ja pienillä kirjaimilla sekä sisennyksillä ja välilyönneillä on merkitystä
- Tietokone ei osaa suoraan ajaa koodia, vaan se on ensin käännettävä kääntäjän (compiler) avulla
- Jotkin kielet, kuten esimerkiksi JavaScript tai Python, suoritetaan niin sanotun tulkin (interpreter) avulla
- Tyypillinen ohjelmakoodi tekee yhden operaation kerrallaan, eli se on yksisäikeistä (single thread)
- Joissakin tilanteissa, tai erikseen pyydettyäessä, samasta ohjelmakoodista voidaan suorittaa useita rinnakkaisia kopioita yhtä aikaa

Ohjelmointikielten käyttöympäristöjä



Python-kieli

- Python on 1980-luvun lopussa kehitetty, avointa lähdekoodia oleva tulkattu (interpreted) ohjelmointikieli, joka sopii moneen tarkoitukseen
- Kieli on dynaamisesti tyyppitetty, ja monesta muusta ohjelmointikielestä poiketen ohjelmakoodin välilyönnit ja sisennykset vaikuttavat koodin suorittamiseen
- Automaattinen muistinhallinta ja roskienkeruu huolehtivat muuttujien ja olioiden muistin vapauttamisesta
- Kaksi pääversiota ovat 2.x ja 3.x, jotka eivät ole keskenään täysin yhteensopivia
 - Yksi käytännön eroista on, että funktioiden kutsut vaativat 3.x-versiossa aina sulut: `print("ABC")`
- Toimii kaikilla käyttöjärjestelmillä, myös erikoisemmilla ja pienemmillä alustoilla (esim. MicroPython)
- Tarkemmin: <https://www.python.org/>



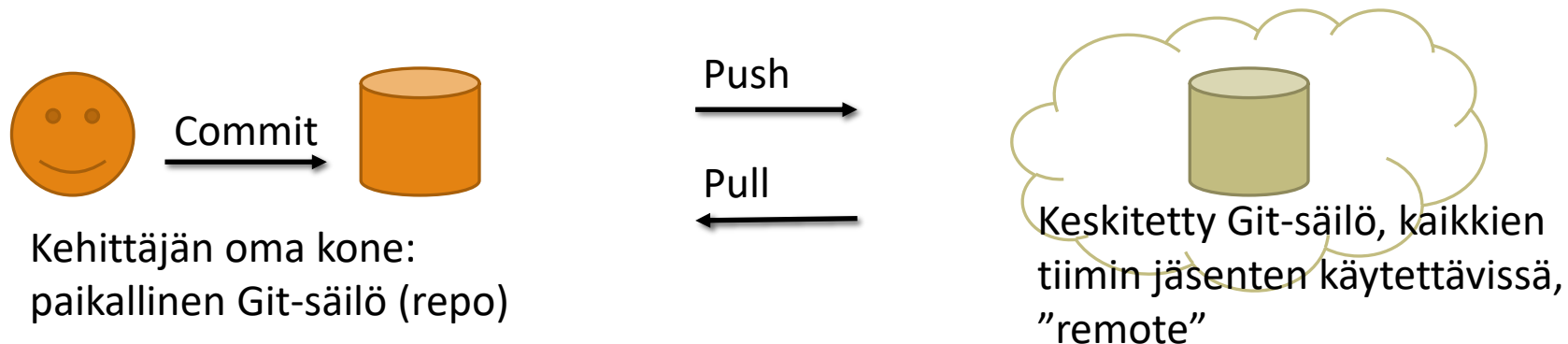
Git-versionhallinta

Git-versionhallinta

- Git on alun perin Linuxin kehittämistä varten suunniteltu versionhallintajärjestelmä, joka on nykyään hyvin suosittu
- Git on lähtökohtaisesti komentorivipohjainen ohjelma, mutta usein sitä käytetään jonkin koodieditorin kautta
- Perustuu hajautettuun (distributed) arkkitehtuuriin
 - Kukin kehittäjä työskentelee oman paikallisen versionhallintansa kanssa, mutta synkronoi tiedot palvelimen kanssa säännöllisesti
- Keskeinen käsite: repository eli repo, suomeksi säilytyspaikka tai säilö
- Latauslinkki
 - <https://git-scm.com/downloads>

Pilvipohjaiset Git-versionhallintapalvelut

- Git-versionhallinta on lähtökohtaisesti aina paikallinen järjestelmä, josta voidaan kuitenkin kopioida tietoja nk. etäpalvelimelle (engl. remote)
- Kun Gitin kanssa tehdään tiimityötä, on erityisesti tärkeää muistaa tallentaa omat muutokset etäpalvelimelle riittävän usein: tähän käytetään push-komentoa
- Markkinoilla on saatavilla useita sekä maksuttomia että maksullisia pilvipalveluita, joita voidaan käyttää Git-versionhallinnan etäpalvelimenä
 - Esimerkiksi GitHub (Microsoft), Bitbucket (Atlassian), Gitlab, Unfuddle, jne.
 - Kurssilla käytämme esimerkkinä GitHubia, <https://github.com/>



Tärkeimmät Git-komennot

- Alustus
 - Init: alustaa hakemiston Git-säilytyspaikaksi
 - Clone: kloonaa eli lataa verkosta jonkin Git-säilytyspaikan tiedot
- Tietojen haku ja lähetys palvelimelle
 - Fetch: hakee palvelimelta tiedot uusimmista tiedostoista
 - Pull: uusimpien tiedostojen lataaminen omalle koneelle
 - Push: ominen muutosten lähettäminen takaisin palvelimelle
- Tallennusten tekeminen paikallisesti
 - Commit: muutosten tallentaminen paikallisesti osaksi versionhallintaa
- Haarat
 - Branch: muodostaa uuden haaran paikallisesti
 - Checkout: vaihtaa haaraa (tämän jälkeen tyypillisesti Pull)
- Tilatieto
 - Status: kertoo, missä tilassa paikalliset tiedostot ovat

Harjoitus: Gitin käyttöä

- Asennetaan Git omalle koneelle
- Perustetaan uusi oma tallennuspaikka (repository)
- Testataan versionhallinnan perustoimintoja

- Vinkki: Windowsissa voit käyttää Posh-Git -nimistä lisäosaa

Tärkeitä Git-taitoja

- Kun Gitin käyttöä opetellaan, on tärkeää hallita vähintään seuraavia perustietoja
- Gitin alustaminen käyttöön
 - Git init -komento
- Olemassa olevan Git-säilön (repon) kloonaaminen omalle koneelle
 - Git clone -komento
- Muutosten tallentaminen omaan Git-tietokantaan
 - Git commit -komento
- Tilan tarkistaminen ja uusien tiedostojen lisääminen
 - Git status- ja Git add -komennot
- Historiatietojen katsominen
 - Git log -komento

JavaScript-kieli

Yleiskuva: JavaScript – lyhyesti JS

- JavaScript on laitteistoriippumaton, yleinen ohjelmointikieli, johon törmää monessa paikassa
- Sopii useisiin käyttötarkoituksiin, ei pelkästään web-sovelluksiin selaimissa
- Ensimmäiset versiot vuoden 1995 paikkeilla
- Keskeiset asiat
 - Syntaksiltaan C-pohjainen kieli, mutta sisältää paljon funktionaalisten kielten (kuten Lisp) ominaisuuksia
 - Melkein olio-pohjainen
 - Prototyyppi-pohjainen
 - Dynaamisesti (heikosti) tyypitetty
- Standardoitu kieli, esim. ECMAScript 2015 -standardi eli ES6 "Harmony"
- Muista: JavaScript on eri asia kuin Java-kieli

JavaScript-esimerkki funktiosta

```
function summa() {  
    var s = 0;  
    for (var i = 0; i < arguments.length; i++) {  
        s += arguments[i];  
    }  
    return s;  
}  
  
summa(1, 2);    // palauttaa 3  
summa(1, 2, 3); // palauttaa 6
```

Missä JavaScript toimii?

- Tyypillisesti JavaScriptiä ajetaan tulkattuna tai JIT-käännettynä
 - Engl. "interpret" tai "just-in-time compilation"
- JavaScript toimii kaikissa web-selaimissa: IE, Edge, Chrome, Firefox, Safari, Opera, ...
- Toimii kaikilla alustoilla, kuten Linux-, Mac- ja Windows-ympäristöissä
 - Erityisesti Node.js on tarkoitettu myös palvelinsovelluksia varten
- Yleisimmät JavaScript-moottorit
 - Mozilla SpiderMonkey
 - Google V8
 - Microsoft Chakra ja JScript

C#- ja JavaScript-kielten eroja

- Vertaillaan edellisellä kurssipäivällä katsotun C#-kielen ja JavaScript-kielen eroja
- Selvitetään esimerkiksi
 - Muuttujien määrittelyä
 - Sovelluksen koodin suorittamista
 - Erilaisia tietotyyppejä
 - Ajonaikaisen ympäristön tarvetta ja kokoa

Lopuksi

Tehtäviä

1. Selvitä, mitkä ovat kolme yleisintä ohjelmointikieltä web-sovellusten kehittämiseen tällä hetkellä.
2. Selvitä, mitä tarkoitetaan käsitteellä *alusta* (engl. platform) web-sovelluksista puhuttaessa?
3. Pohdi, mitä eroja ja samankaltaisuuksia tietokoneiden ohjelmointikielillä on ihmisten käyttämiin kieliin.
4. Git on yksi tämän hetken suosituimmista versionhallintajärjestelmistä. Mitä muita versionhallintajärjestelmiä on olemassa? Nimeä ainakin kolme.
5. Listaa syitä, miksi versionhallinta on hyödyllinen, vaikka koodaisit sovelluksia vain yksin?

Koodausta

1. Asenna koneellesi Node.js-ympäristö osoitteesta www.nodejs.org. Millä komentorivikomennolla saat selville, mitkä versiot koneellasi on Nodesta ja NPM:stä?
2. Kirjoita Hello World -sovellus JavaScriptillä, ja aja se Node.js-ympäristössä.
3. Vertaile aiemmin tekemääsi C#-kielistä Hello World -ohjelmaa JavaScript-versioon. Mitä samankaltaisuuksia ja eroavaisuuksia löydät? Kumpi ohjelma on nopeampi suorittaa?
4. Alusta itsellesi paikallinen Git-versionhallintakansio. Kopioi sitten kansioon tekemiäsi ohjelmakooditiedostoja ja tee ensimmäinen tallennus versionhallintaan (commit).