

Visual Studio Code -extensionit

- Jos käytät VSCodea, saatat tykätä seuraavista lisäosista
- C#
 - <https://marketplace.visualstudio.com/items?itemName=ms-dotnettools.csharp>
- VS Sharper for C#
 - <https://marketplace.visualstudio.com/items?itemName=eservice-online.vs-sharper>



240

C# konsoliprojektin luominen

- Ensimmäisenä, avaa terminaali/konsoli/powershell/cmd ja mene jonnekin sopivaan paikkaan hakemistorakenteessasi
 - Sopiva paikka on esimerkiksi hakemisto, jossa säilytät muita tähän opintojaksoon liittyviä materiaaleja
- Konsoli/terminaalikomento
 - `dotnet new console -o <nimi>`
 - Korvaa <nimi> jollakin kuvaavalla nimellä
- Luo projektin uuteen kansioon jonka nimi on <nimi>
 - Kansio luodaan siihen sijaintiin, missä olet komentoa suorittaessa



241

Program.cs -tiedosto

```
// See https://aka.ms/new-console-template for more information  
Console.WriteLine("Hello, World!");
```



242

Rakennuselementit

- Muuttujat
- Funktiot
- Ohjausrakenteet
 - Loopit
 - Ehdot



243

Muuttujat

PIENIN LIIKKUVA OSA KOODIA



244

Muuttujat

- Muuttuja on mekanismi jonkin arvon tallentamiseen
- Sillä **kuvaava** on nimi
- Se on riippuvainen ajasta ja paikasta
- Sen arvoa voidaan muuttaa
- Eng. variable



245

Muuttujat

- Muuttujan tyyppi kertoo millainen arvo siihen voidaan tallentaa
- Esim
 - Kokonaisluku
 - Desimaaliluku
 - Merkkijono
 - Totuusarvo
 - Olio



246

Tyypit

- Perustyypit:
 - Kokonaisluvut (sbyte, byte, short, ushort, int, uint, long, ulong)
 - Desimaaliluvut (decimal, float, double)
 - Totuusarvo (bool, Boolean)
 - Merkkityyppi (char)
 - Merkkijono (string)



247

Tyypit

- Luokka
 - Itse ohjelmoitu luokka
- Eri listatyypppejä
 - Array
 - List
 - Collection
 - Queue



248

Muuttuja, sijoitusoperaatio

Arvon tallentaminen muuttujaan

- **tyyppi nimi = lauseke**
- **= on sijoitusoperaattori**
 - Luetaan "saa arvokseen"
 - Esim. `int x = 42` luetaan "kokonaislukumuuttuja x saa arvokseen kokonaisluvun 42"
- Mahdollinen aikaisempi arvo korvataan uudella arvolla, joka saadaan evaluoimalla lausekkeen arvo
 - Lause evaluoidaan aina ensin, sitten vasta tallennetaan muuttujaan



249

Syntaksi

- Muuttujaa määritellessä syntaksi (kirjoitusasu) on
 - `muuttujan_tyyppi muuttujan_nimi = lauseke;`
- Puolipiste (;) lauseen lopussa on tärkeää, se ilmaisee tietokoneelle, että tässä loppuu yksi komento



250

Kokonaisluku

- Englanniksi integer
- Luonnolliset luvut
- C#:ssa tyyppimääritelmä `int`

```
// Asetetaan muuttujan 'age' arvoksi kokonaisluku 35
int age = 35; // Huomaa tyypitys

// Muutetaan muuttujan 'age' arvoksi 20
// Muuttujan tyyppi on jo määritelty,
// joten tyyppiä ei tässä kohtaa sanota.
age = 20;

Console.WriteLine(age);
```



251

Muuttujan arvon tulostaminen

- Käyttämällä komentoa `Console.WriteLine(muuttuja)` saadaan tulostettua muuttujan arvo

```
Console.WriteLine(age);
```



252

Aritmeettiset operaatiot

- Plus, +
- Miinus, -
- Jako, /
- Kerto, *
- Jakojäännös, %

```
int first_number = 76;  
int second_number = 12 * 2;  
  
// Sijoitusoperaattorin (=) oikea puoli (lause) evaluoidaan  
// ensin, joten 'sum' saa arvokseen kokonaisluvun 100.  
int sum = first_number + second_number;  
Console.WriteLine(sum);
```



253

Desimaaliluvut

- Tyypinä float tai double
- Erona näissä on tarkkuus (== varatun muistin määrä)
- Useimmissa tapauksissa käytettävällä tyypillä ei ole merkitystä

```
// Desimaaliluku
float pi = 3.14f;
Console.WriteLine(pi);

// Desimaaliluku
double pi2 = 3.14;
Console.WriteLine(pi2)
```



254

Merkkijono, string

- Tekstin pitämiseen muistissa
- C#:ssa käytetään lainausmerkkien kanssa

```
// Muuttujan 'name' tyyppi on merkkijono
string name = "SAMK";

Console.WriteLine(name);
```



255

Merkkijonojen yhdistäminen

- C# osaa yhdistää (concatenate) kaksi merkkijonoa yhteen + -operaattorin avulla
- Huomataan, että operaattoreiden toiminta voi muuttua annettujen tyyppien mukaan

```
string firstname = "John";  
string lastname = "Doe";  
  
string fullname = firstname + " " + lastname;  
Console.WriteLine(fullname);
```



256

Totuusarvot

- Boolean
- C#:ssa tyyppinä bool
- Arvo voi olla vain true tai false

```
// Totuusarvo  
bool isAlive = true;  
isAlive = false;  
Console.WriteLine(isAlive)
```



257

Tyyppiyhteensopivuus

- Muuttujan ja sijoitettavan arvon tyyppien pitää olla yhteensopivia
- Esimerkiksi kokonaislukutyypin (int) muuttujaan voi sijoittaa vain kokonaislukuja

```
// Kokonaisluku
int a = 3;
// Aiheuttaa virheen, koska tyytit eivät täsmää
a = "3.14";
```