

Java (icd0019)

Märt Kalmo

# „Hello world” Püütonis

```
print("Hello!")
```

# „Hello world” Javas

```
public class Hello {  
  
    public static void main(String[] args) {  
  
        System.out.println("Hello!");  
  
    }  
  
}
```

# „Hello world” Javas

```
  ① public class ② Hello { ③  
    ④ public static void ⑤ main(String[] args) { ⑦ ⑧  
        ⑨ System.out.println("Hello!"); ⑩  
    }  
}
```

# Lausete eristamine

```
public class Hello{public static void  
main(String[]args){System.out.println("Hello!");}}
```

NB! Nii ei tohi teha!

# „Hello world” Javas

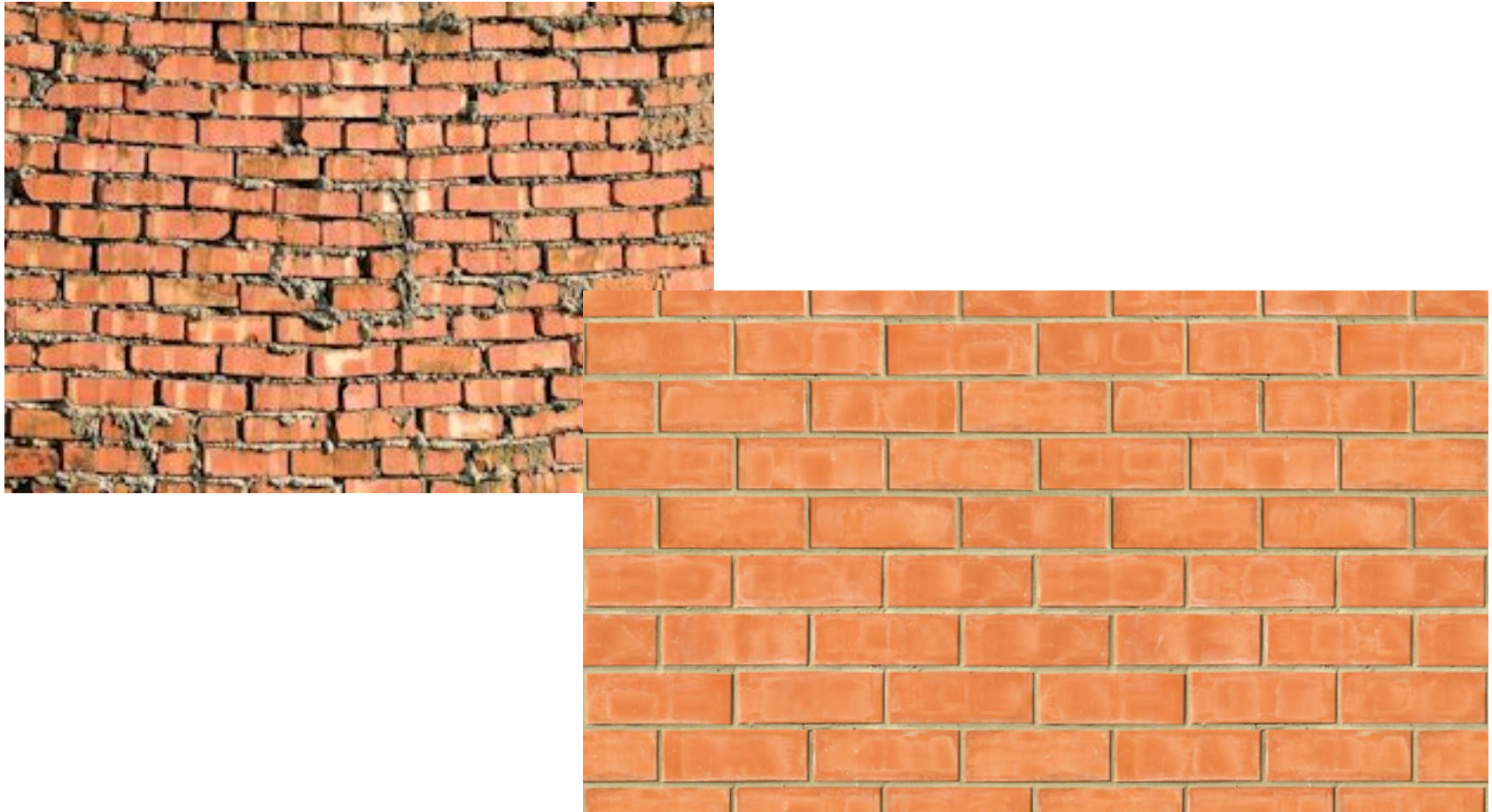
```
  ①      ②      ③
public class Hello {

    ④      ⑤      ⑥      ⑦      ⑧
    public static void main(String[] args) {

        ⑨
        System.out.println("Hello!"); ⑩
    }
}
```

# ITI0102 vs. ICD0019

# Analoogia





# „Hello world” Javas

```
public class Hello {  
  
    public static void main(String[] args) {  
  
        System.out.println("Hello!");  
  
    }  
  
}
```

# Analoogia Püütonis

```
class Hello:  
    def main(self):  
        print("Hello!")
```

# Analoogia Püütonis

```
def fahrenheit(temp_in_celsius):  
    return (temp_in_celsius * 9 / 5) + 32  
  
print(fahrenheit(22))
```

# Analoogia Püütonis

```
class Program:
    def main(self):
        print(self.fahrenheit(22))

    def fahrenheit(self, temp_in_celsius):
        return (temp_in_celsius * 9 / 5) + 32
```

```
Program().main()
```

# Programmi argumendid

- > `git config --global user.name "Taavi Tudeng"`
- > `git config --global user.email uniid@ttu.ee`

# Programmi argumendid

> python my-prog.py

> python my-prog.py arg1 arg2 arg3

# „Hello world” Javas

```
  ① public class ② Hello { ③  
    ④ public static void ⑤ main(String[] args) { ⑦ ⑧  
        ⑨ System.out.println("Hello!"); ⑩  
    }  
}
```

# Muutujad Püütonis

x = 1

...

...

...

x = "Alice"



# Muutujad Javas

```
int x = 1;
```

```
x = 2; // ok
```

```
...
```

```
...
```

```
...
```

```
int x = 2; // error!!!
```

# Tüübid Püütonis

```
x = 1
```

```
name = "Alice"
```

```
x = "Alice"
```

# Tüübid Javas

```
int x = 1;
```

```
String name = "Alice";
```

```
x = "Bob"; // error!!!
```

# Termin

- Andmetüüp (nt. täisarv, string, boolean, jne)
- nt. „age” on täisarv tüüpi muutuja

# Objekt vs klass

```
class Person:  
    def __init__(self, name):  
        self.name = name
```



```
alice = Person("Alice")  
bob = Person("Bob")
```

# Termin

- Isend (objekt mingist klassist)

# Objekt vs klass

```
public class Person {  
    private String name;  
  
    public Person(String name) {  
        this.name = name;  
    }  
}
```



```
Person alice = new Person("Alice");  
Person bob = new Person("Bob");
```

# Termin

- Andmetüüp (nt. täisarv, string, **Person**, jne)
- nt. „alice” on **Person** tüüpi muutuja



# Tüübid Javas

```
public class Hello {  
  
    public static void main(String[] args) {  
        int sum = add(1, 2);  
    }  
  
    public static int add(int x, int y) {  
        return x + y;  
    }  
  
}
```

# Tüübid Javas

```
public class Hello {  
  
    public static void main(String[] args) {  
        System.out.println("Hello!");  
  
        int sum = printHello(); // error !!!  
    }  
  
    public static void printHello() {  
        System.out.println("Hello!");  
    }  
  
}
```

# Staatiline analüüs

# „Hello world” Javas

```
  ① public class ② Hello { ③  
    ④ public static void ⑤ main(String[] args) { ⑦ ⑧  
        ⑨ System.out.println("Hello!"); ⑩  
    }  
}
```

# Koodi käivitamine Püütonis

> python hello.py

# Koodi käivitamine Javas

> `javac Hello.java`

Selle tulemusena tekib fail `Hello.class`

> `java Hello`

Käivitab klassi `Hello` `main()` meetodis oleva koodi

# Miks nii keeruline?

# Miks nii keeruline?

- Jõudlus
- Varajane vigade avastamine
- Selgem käitumise määramine
- Koodi hallatavus



# Masinkood

- hello.exe

# Interpretaator (Püüton)

- `python prog.py == python.exe prog.py`

prog.py

```
var = 3

numbers = [1, 2, var]

numbers.reverse()

for each in numbers:
    print(each)
```

# Kompileerimine (C)

hello.c -> ... -> hello.exe

> hello.exe

# Kompilieren (Java)

Hello.java -> Hello.class

> java Hello

# Varajane vigade avastamine

```
def add(x, y):  
    return x + y
```

```
a = 1
```

```
if z < 2:  
    a = 'üks'
```

```
print(add(a, 1))
```

# Varajane vigade avastamine

- Vigade otsimine jaguneb kaheks etapiks

# Selgem käitumise määramine

```
print(6 * 0.1)    # 0.6000000000000000000001
```

# Koodi hallatavus

- Olemasoleva koodi muutmine
- Koodist arusaamine
- Koodi ümberpaigutamine



# Koodi hallatus (Python)

main.py

```
from other import *  
  
hello()
```

other.py

```
def hello():  
    print("Hello from other.py")
```

# Koodi hallatus (Python)

main.py

```
if a == 2:  
    from other import *  
  
if a > 2:  
    from third import *  
  
hello()
```

other.py

```
def hello():  
    print("Hello from other.py")
```

# Koodi hallatavus (Java)

- Staatile analüüs annab kindla vastuse
- Väga hea tööriistade toetus
  - koodi leidmine
  - koodi automaatne ümbertõstmine

# Fakt

- Valdav enamus Eesti suurematest infosüsteemidest on kirjutatud Javas või C#-is.

# Kursusest

# Hinne

- 51: 1, 61: 2, ... 91: 5
- Eksam kuni 80 punkti
- Iseseisvad ülesanded kuni 20 punkti

# Iseseisvad ülesanded

- Individuaalsed
- Esitamine kaheteistkümnes osas

# Materjalid

- Loengud
- Harjutustunnid
- YouTube või Echo360 salvestused



# Harjutustunnid

- Loengu materjali kinnistamiseks
- Seletused ekraanivideona

# 1. ja 2. Harjutustund

- N 10:00
  - IADB21, IADB22, IAAB41, IAAB42 ja kõik teised
- N 12:00
  - IADB23, IADB24, IADB25

# Materjalid

- Slaidid on loengut illustreerivad materjalid ja ei ole mõeldud konspektiks.
- Konspektiks võiks olla lahendatud harjutstundide ülesanded.

# Ülesannete esitamine

- Esitamine läbi BitBucket-i ([bitbucket.org](https://bitbucket.org))
- Repositoorium on **privaatne**
- Kõik osad lähevad samasse repositooriumi
- Esitav commit märgistada (tag)
- Esitatud töö peab käivituma ja läbima testid
- Automaatne kontroll ja võimalus parandada

# Projekti esitamine

The screenshot shows a web browser window with the Bitbucket interface. The address bar displays the URL `https://bitbucket.org/mkalmo2/icd0019/commits/all`. The page title is "Märt Kalmo 2 / icd0019" and the main heading is "Commits". A sidebar on the left contains navigation icons, including a "java" icon. A table of commits is displayed with columns for Author, Commit, Message, Date, and Builds. The commits are listed in descending order of time, with the most recent at the top. A search bar labeled "Find commits" is located in the top right of the commit list area. A vertical commit history graph is visible on the left side of the commit table.

Author	Commit	Message	Date	Builds
Märt Kalmo	e215f46	test	18 hours ago	
Märt Kalmo	bccc6dc	Merge remote-tracking branch 'origin/master' # Conflicts: # src/Hell...	18 hours ago	
Märt Kalmo	6798699	modified hello.java	18 hours ago	
Märt Kalmo 2	aae8e63	Hello.java edited online with Bitbucket	18 hours ago	
Märt Kalmo	a1b287d	changed pass	18 hours ago	
Märt Kalmo	304eeed	modified info.json	18 hours ago	ex01
Märt Kalmo	bb046a0	modified info.json	18 hours ago	
Märt Kalmo	3c7e2c7	modified info.json	21 hours ago	
Märt Kalmo	10b326f	1st	23 hours ago	

# Ülesannete hindamine

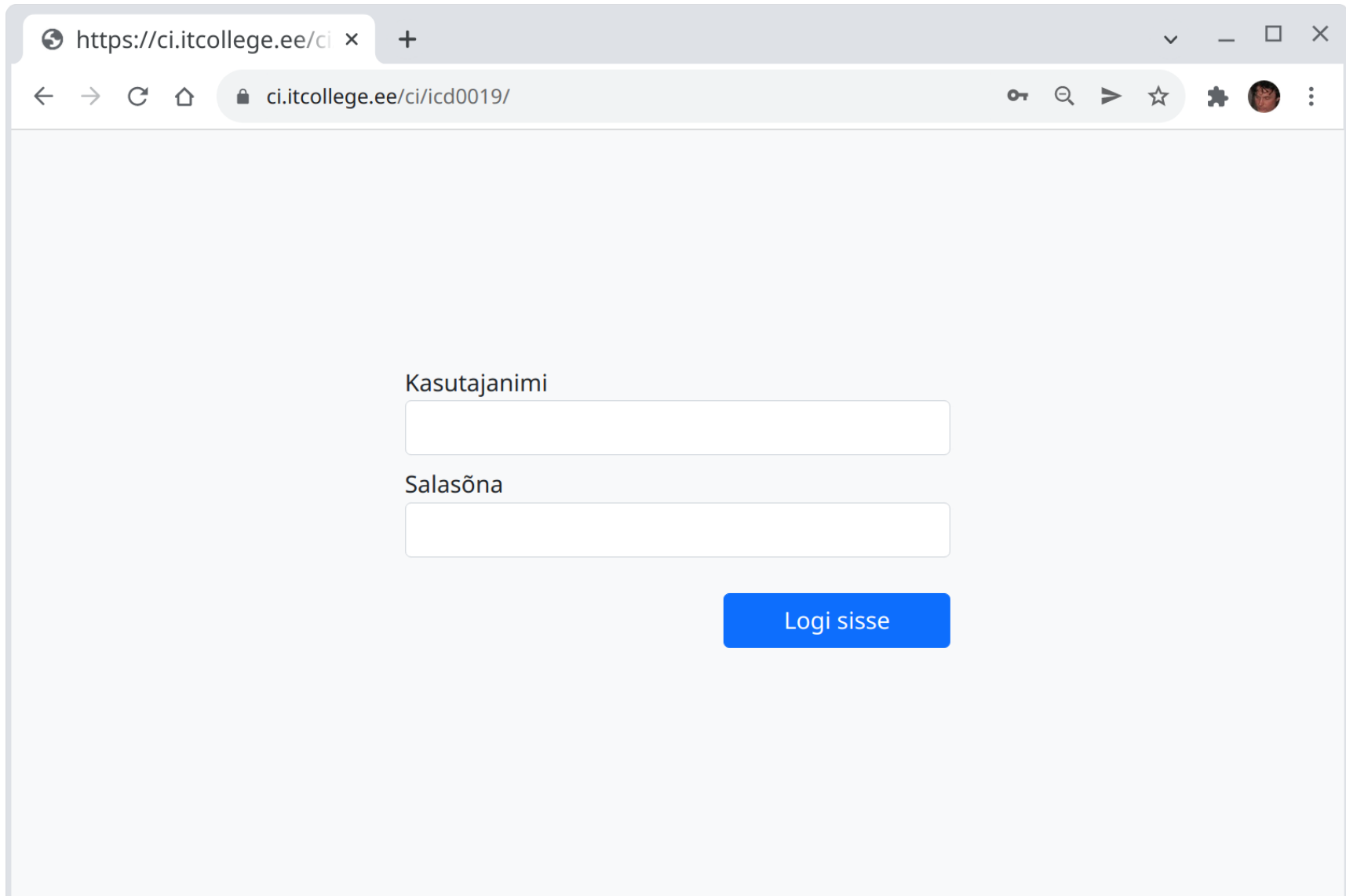
- Ülevaatamine on poolautomaatne

# Tagasiside

- Individuaalset tagasisidet oma tööle saate küsida harjutustunni ajal.

# Hindamissüsteem

<https://ci.itcollege.ee/ci/icd0019>



The screenshot shows a web browser window with the address bar displaying <https://ci.itcollege.ee/ci/icd0019/>. The page content is a login form with the following elements:

- A label "Kasutajanimi" (Username) above a text input field.
- A label "Salasõna" (Password) above a text input field.
- A blue button labeled "Logi sisse" (Login) positioned below the password field.



# Hindamissüsteem

https://ci.itcollege.ee/ci x +

← → ↻ 🏠 🔒 ci.itcollege.ee/ci/icd0019/results/mkalmo2 🔍 ▶ ☆ ⚙️ 👤 ⋮

Projekt Ülevaatused

Logi välja

Projekti tulemused

Tag	Räsi	Punktid	Tähtaeg	Läbitud	Viimati käivitatud
ex01	ec0f9ff	<u>3/5</u>	03. Feb (22:00)	25. Jan (15:35)	25. Jan (15:35)
ex02			10. Feb (22:00)		
ex03			17. Feb (22:00)		
ex03a			23. May (22:00)		
ex04			24. Feb (22:00)		
ex05			03. Mar (22:00)		
ex06			10. Mar (22:00)		
ex06a			23. May (22:00)		
ex07			17. Mar (22:00)		
ex07a			23. May (22:00)		
ex08			24. Mar (22:00)		

# Hindamissüsteem

https://ci.itcollege.ee/ci x +

ci.itcollege.ee/ci/icd0019/output-log/mkalmo2/ex01

Projekt Ülevaatused Logi välja

Tagasi

----- Test output -----

nameIsWrittenAsInOis(intro.JsonTests) failed  
java.lang.AssertionError: There is no declaration with this name in Õis (as of 25.01). If you declared the course later you will get the points at 14.02  
intro.JsonTests.nameIsWrittenAsInOis(JsonTests.java:76)

convertsDecimalToBinaryAndBack(intro.BinaryConverterTests) failed  
java.lang.AssertionError: converting  
Expected: is <1>  
but: was <0>  
intro.BinaryConverterTests.convertsDecimalToBinaryAndBack(BinaryConverterTests.java:20)

# Koodi stiili kontroll

- Esitatud ülesanded peavad läbima stiilikontrolli

# Eksam

- Eeldus: kõigi nõutud ülesannete esitamine

# Tähtajad

- Iga ülesannete komplekti esitamiseks on määratud tähtaeg.
- Pärast tähtaega esitatud ülesannete eest punkte ei saa

# Tähtajad

- Pärast 23. maid ühtegi esitamist ei arvestata.

# Reeglid

Course: ICD0019 Java

moodle.taltech.ee/course/view.php?id=24943

Veeb ÕIS Moodle E-mail

TAL TECH Moodle Help English (en)

Märt Kalmo  
Õppija/Student

- REEGLID JA TÄHTAJAD
- Tulemuste vaatamine
- Sissejuhatus kursusele
- Ainekava (ÕIS)
- E-tugi
- Kursuse foorum

Et õppejõud küsimusele vastaks peaks see olema võimalikult konkreetne, et sellele saaks mõne lausega vastata. Küsimustele (väidetele) "mul programm ei tööta" või "ei saanud millestki aru" ei saa kirjalikult vastata. Kui teil selline mure on, siis peaksite abi saamiseks harjutustundi tulema.

# Võimalikud erandid

- Reeglitesse ei tehta isiklikke erandeid.



# Koodi ülevaatused

# Mahakirjutamine

- Mahakirjutamine = mahakirjutamine eksamil
- Automaatkontroll

# Mahakirjutamine

- Git-i repot ei tohi kustutada
- Kopeerimisel peab allikale viitama
- Töö mahakirjutamiseks andmine on täpselt sama moodi petmine
- Individuaalset projekti ei tehta koos

# Kontakt

# Kontakt

- Kursuse foorum Moodles
- [mart.kalmo@taltech.ee](mailto:mart.kalmo@taltech.ee)
- Kirja teel ei vasta vea otsimise küsimustele

# Tagasisidest

- Slaididest saab ainult loengutes aru, need on täiesti kasutud kui puudud tunnist

# Tagasisidest

- Nimelt mina isiklikult üldjuhul neid videosid ei vaadanud (sest ma jänni eriti tihti ei jäänud) aga mis mulle hiljem pähe tuli, oli see, et ma kujutan ette, et nii mõndagi ülessannet võisin mina veidi teistmoodi lahendada kui videos (ja ma poleks sellest teada saanudki, sest ma videot ei taibanud vaadata). Kui ma lahendasin lihtsalt teistmoodi, siis pole hullu aga **kui ma ikka mingit rämedat kilplase tööd tegin**, siis see ka nagu eriti hea ei ole. ...

# Tagasisidest

- Õpetajal ükskõikne suhtumine kuidas ta õpilastel läheb



# Kursuseks vajalik tarkvara

- Java (17)
- IntelliJ IDEA
- Git

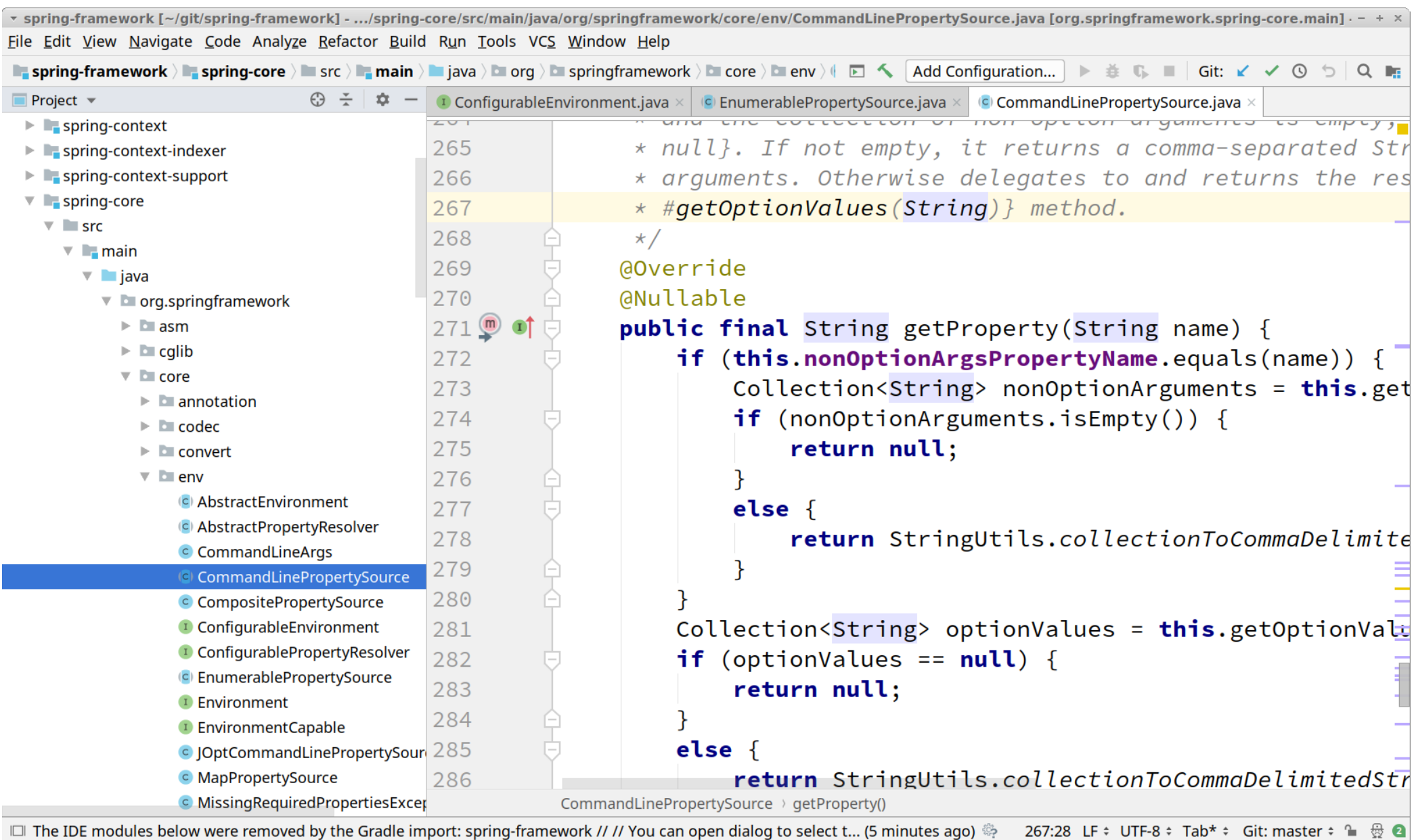
# IDE

- Integrated development environment
- nt. IntelliJ IDEA, Eclipse, NetBeans

# IDE

- Koodi kompileerimine
- Koodis navigeerimine
- Koodist otsimine
- Sõltuvuste leidmine
- Versioonihaldus
- Silumine
- Üksustestide käivitamine
- jne ...

# IDEA



# IntelliJ IDEA

- <https://www.jetbrains.com/idea/download>  
(soovitav on Ultimate versioon)
- <https://www.jetbrains.com/student/>