LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Lehrstuhl Bioinformatik • Konstantin Pelz

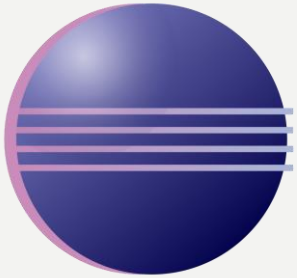
Erste Java-Programme

(Eclipse und primitive Datentypen)

Tutorium Bioinformatik
(WS 18/19)

Konstantin: Konstantin.pelz@campus.lmu.de
Homepage: https://bioinformatik-muenchen.com/studium/propaedeutikum-programmierung-in-der-bioinformatik/

Eclipse



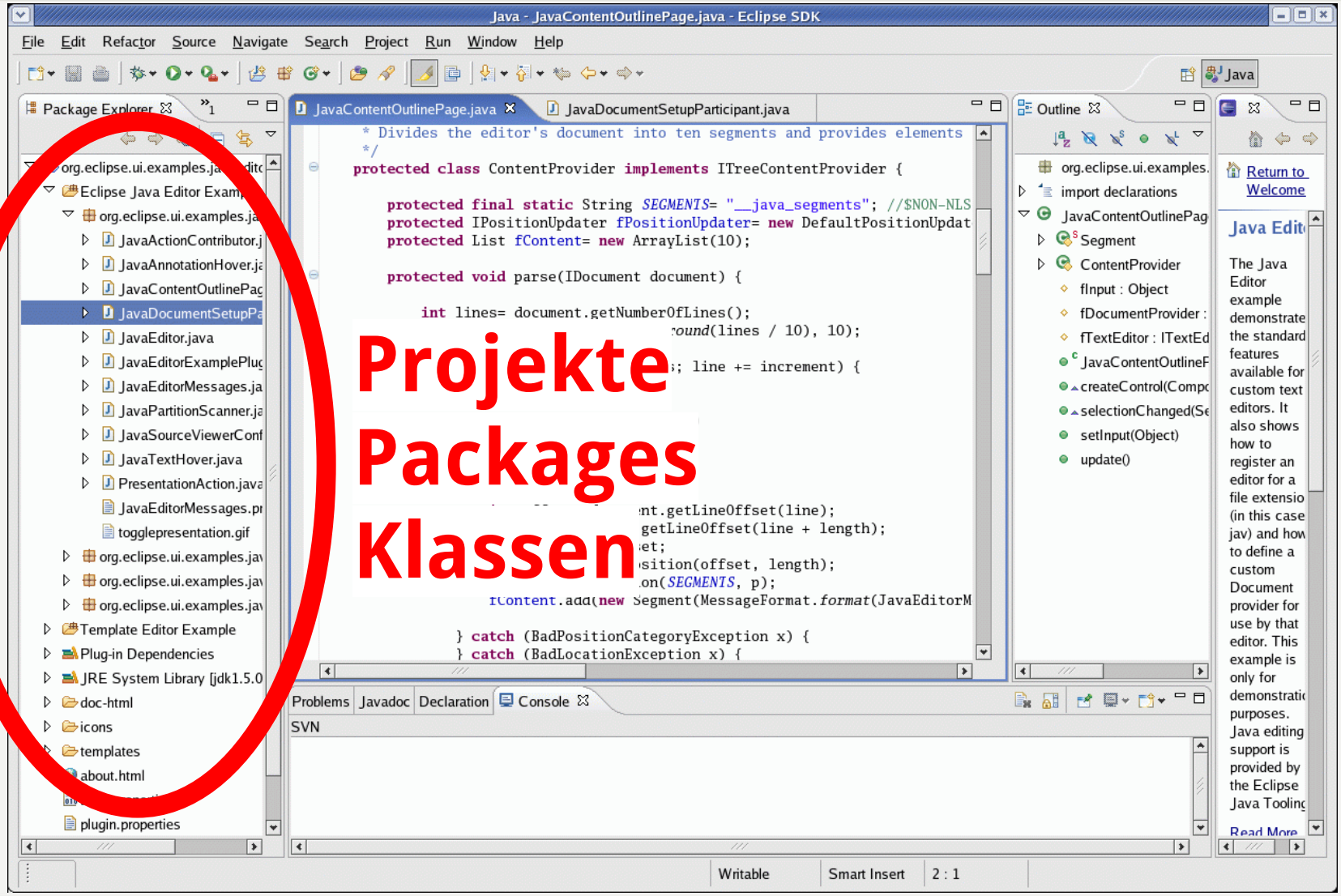
IDE: integrated development environment

- **Texteditor**
- **Compiler bzw. Interpreter**
- **Debugger**
- **Quelltextformatierung**

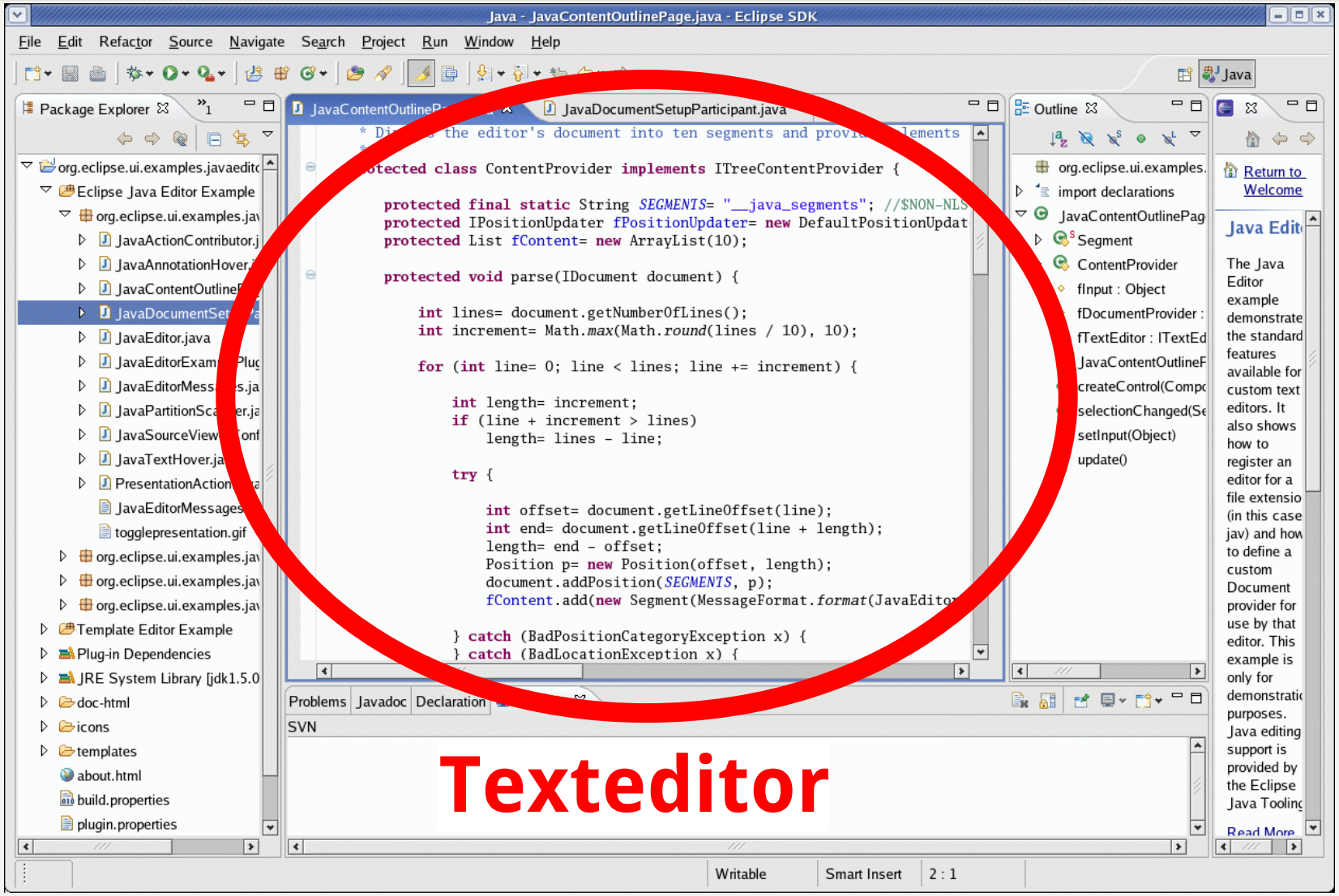
Download:
https://eclipse.org/downloads/

CIP:
/home/proj/software/eclipse/

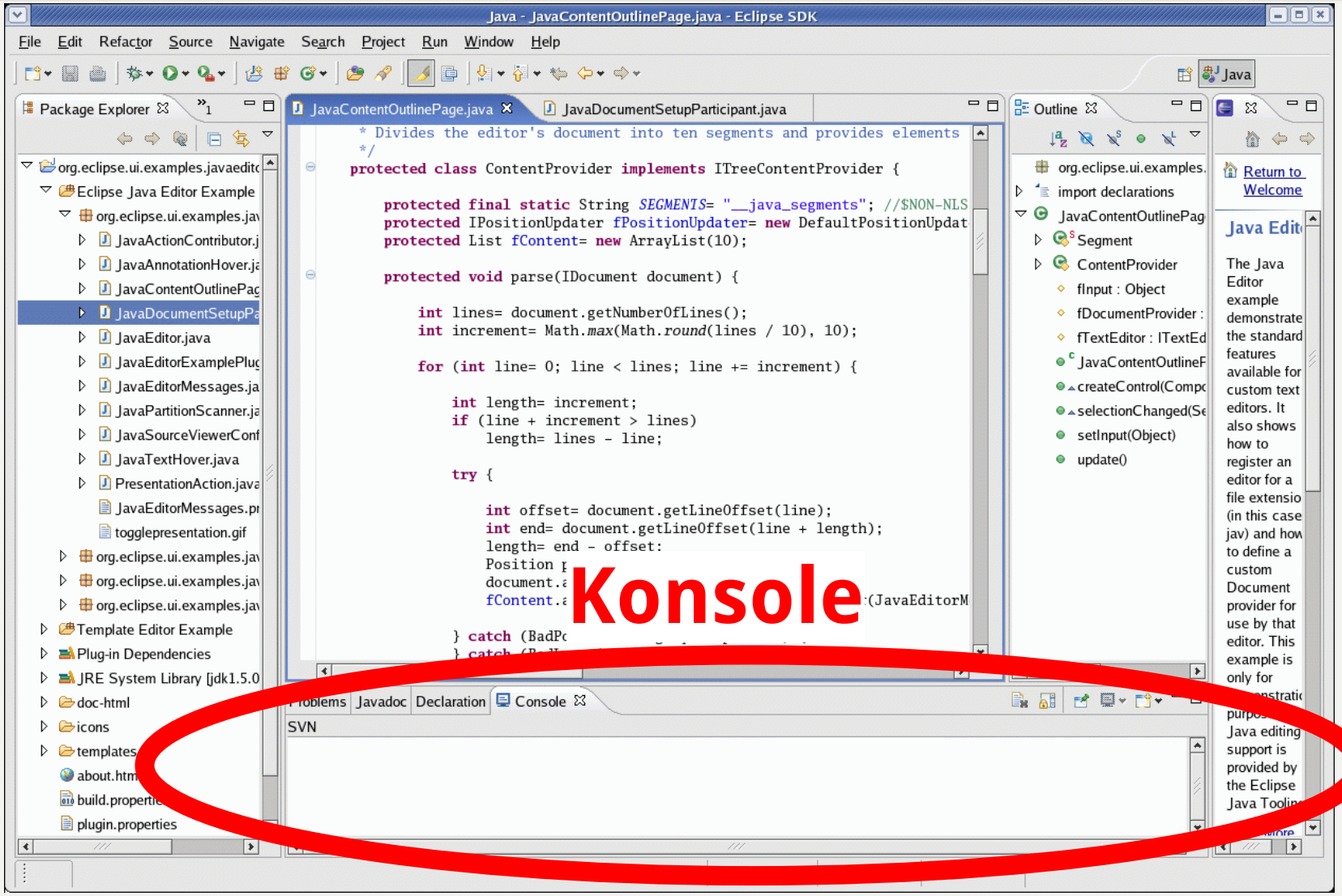
Eclipse


Java - JavaContentOutlinePage.java - Eclipse SDK
File Edit Refactor Source Navigate Search Project Run Window Help
Package Explorer
JavaContentOutlinePage.java
JavaDocumentSetupParticipant.java
Outline
Problems Javadoc Declaration Console
SVN
Writable
Smart Insert
2 : 1
Projekte
Packages
Klassen

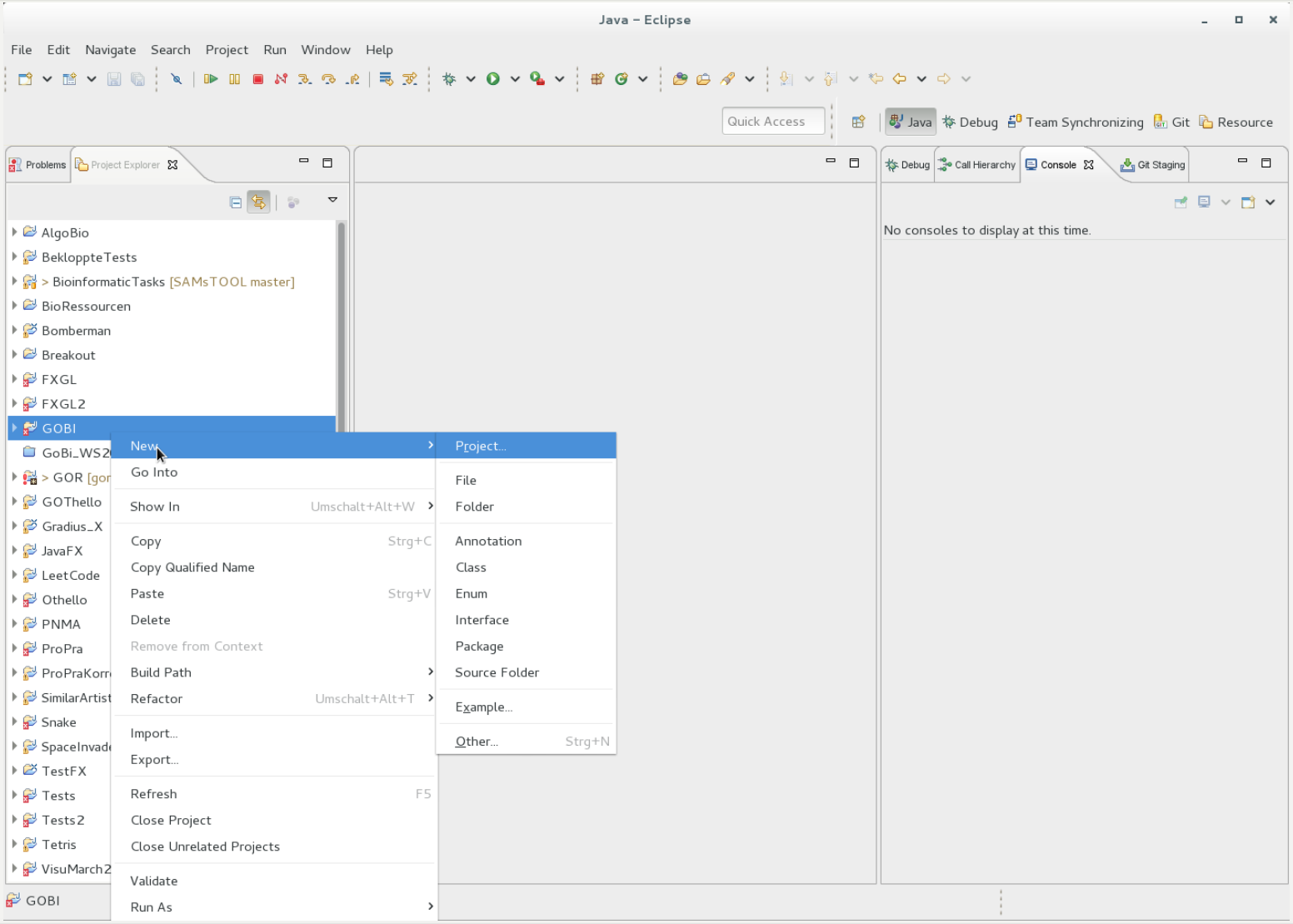
Eclipse


Java - JavaContentOutlinePage.java - Eclipse SDK
File Edit Refactor Source Navigate Search Project Run Window Help
Package Explorer
JavaDocumentSetupParticipant.java
Outline
Problems Javadoc Declaration
SVN
Writable
Smart Insert
2 : 1
Texteditor

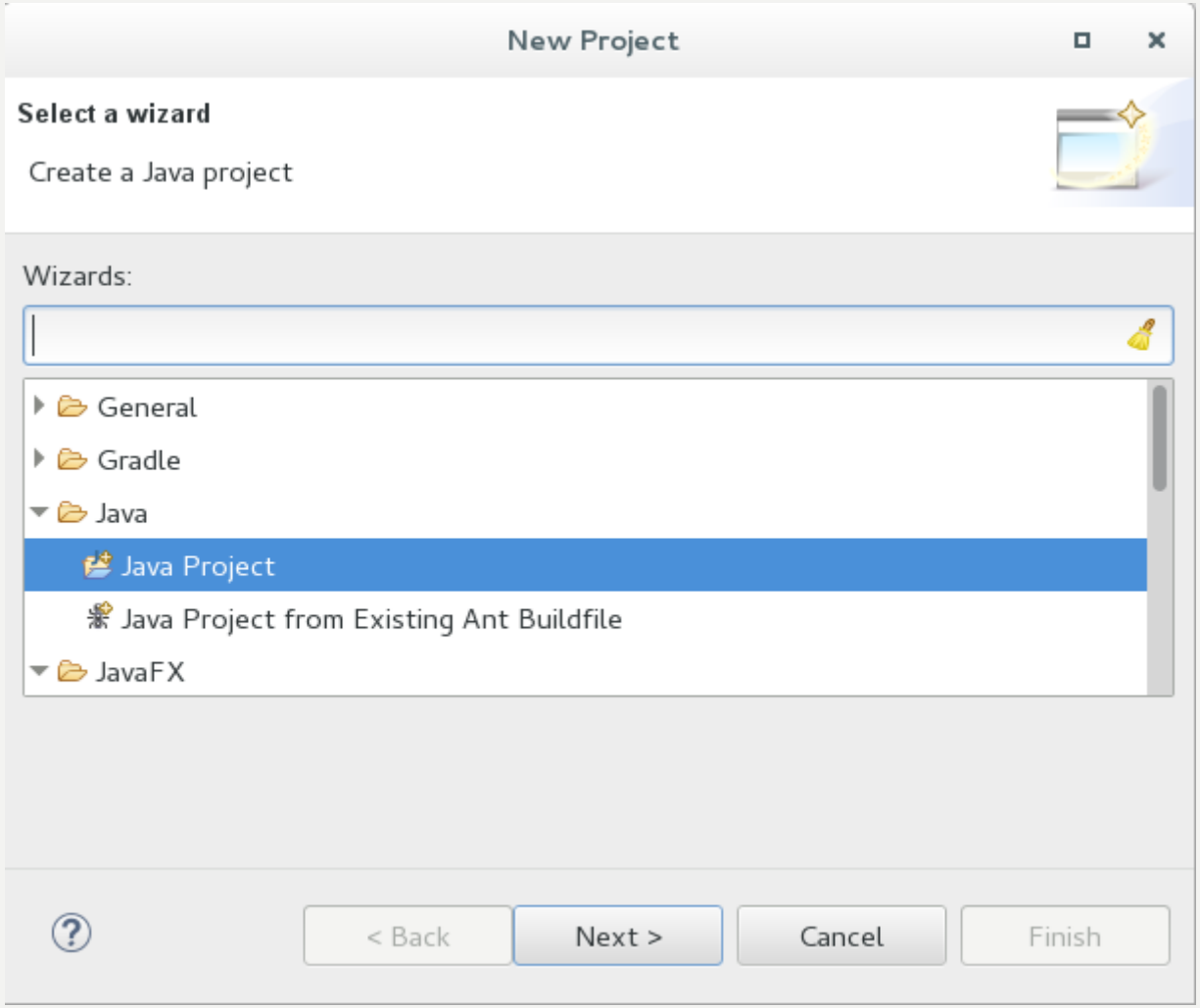
Eclipse


Java - JavaContentOutlinePage.java - Eclipse SDK
File Edit Refactor Source Navigate Search Project Run Window Help
Package Explorer
JavaContentOutlinePage.java
JavaDocumentSetupParticipant.java
Outline
Problems Javadoc Declaration Console
SVN
Konsole

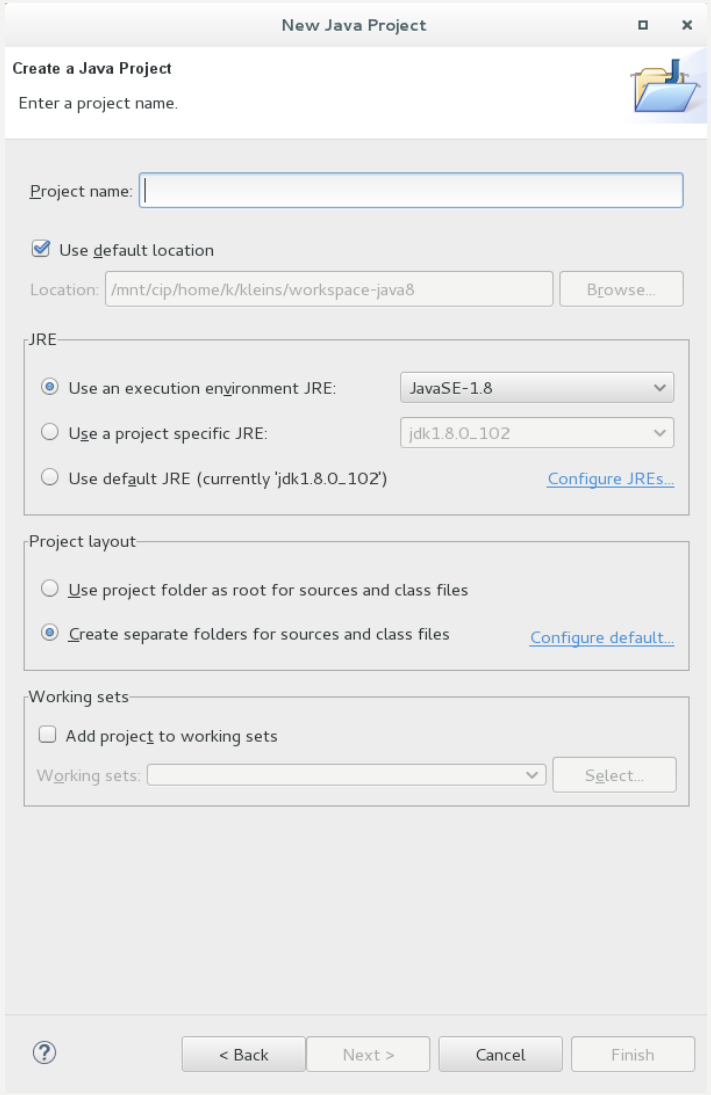
Eclipse


Java – Eclipse
File Edit Navigate Search Project Run Window Help
Quick Access
Java Debug Team Synchronizing Git Resource
Problems Project Explorer
AlgoBio
BekloppteTests
> BioinformaticTasks [SAMsTOOL master]
BioRessourcen
Bomberman
Breakout
FXGL
FXGL2
GOBI
GoBi_WS2
> GOR [gor
GOThello
Gradius_X
JavaFX
LeetCode
Othello
PNMA
ProPra
ProPraKorr
SimilarArtist
Snake
SpaceInvad
TestFX
Tests
Tests2
Tetris
VisuMarch2
GOBI
New
Go Into
Show In Umschalt+Alt+W
Copy Strg+C
Copy Qualified Name
Paste Strg+V
Delete
Remove from Context
Build Path
Refactor Umschalt+Alt+T
Import...
Export...
Refresh F5
Close Project
Close Unrelated Projects
Validate
Run As
Project...
File
Folder
Annotation
Class
Enum
Interface
Package
Source Folder
Example...
Other... Strg+N
Debug Call Hierarchy Console Git Staging
No consoles to display at this time.

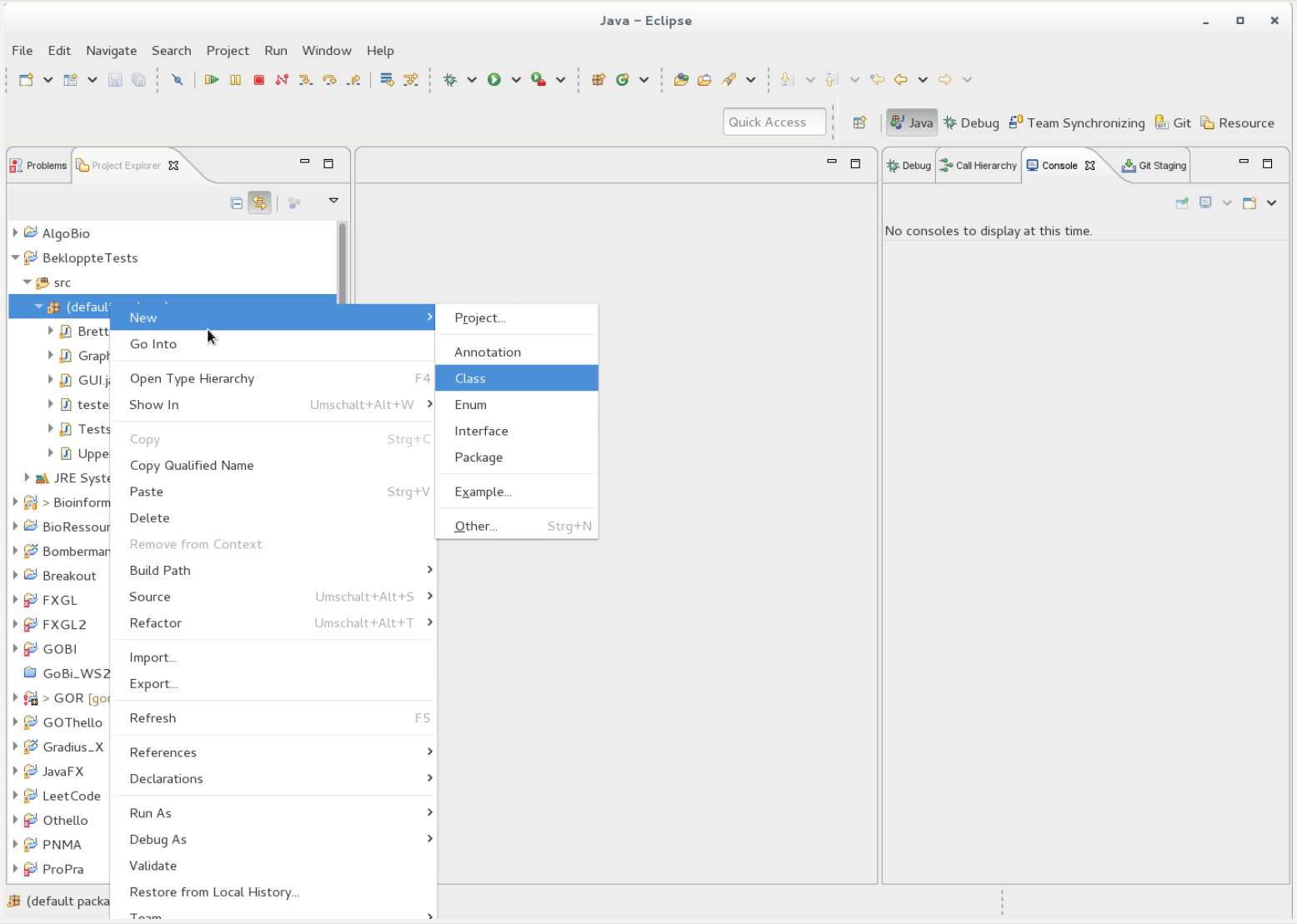
Eclipse


New Project
Select a wizard
Create a Java project
Wizards:
General
Gradle
Java
Java Project
Java Project from Existing Ant Buildfile
JavaFX
< Back
Next >
Cancel
Finish

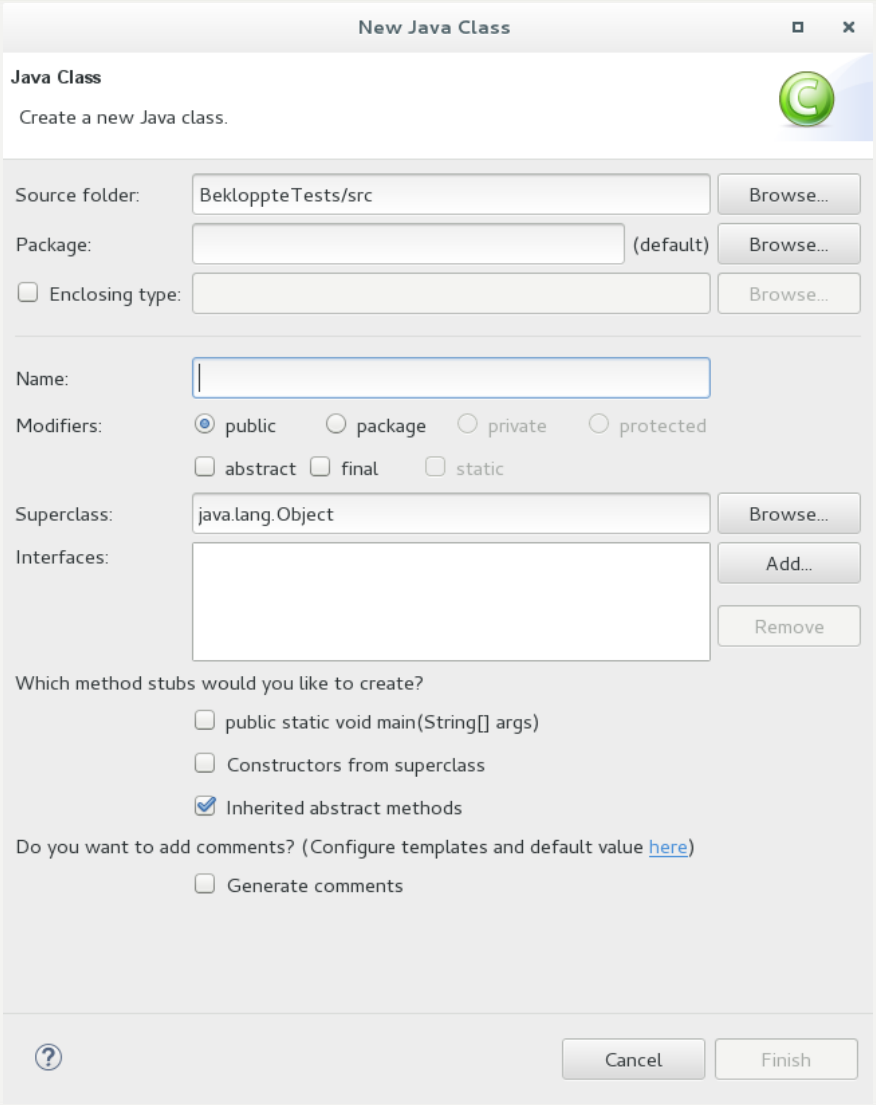
Eclipse


New Java Project
Create a Java Project
Enter a project name.
Project name:
Use default location
Location: /mnt/cip/home/k/kleins/workspace-java8
Browse...
JRE
Use an execution environment JRE: JavaSE-1.8
Use a project specific JRE: jdk1.8.0_102
Use default JRE (currently 'jdk1.8.0_102')
Configure JREs...
Project layout
Use project folder as root for sources and class files
Create separate folders for sources and class files
Configure default...
Working sets
Add project to working sets
Working sets:
Select...
< Back
Next >
Cancel
Finish

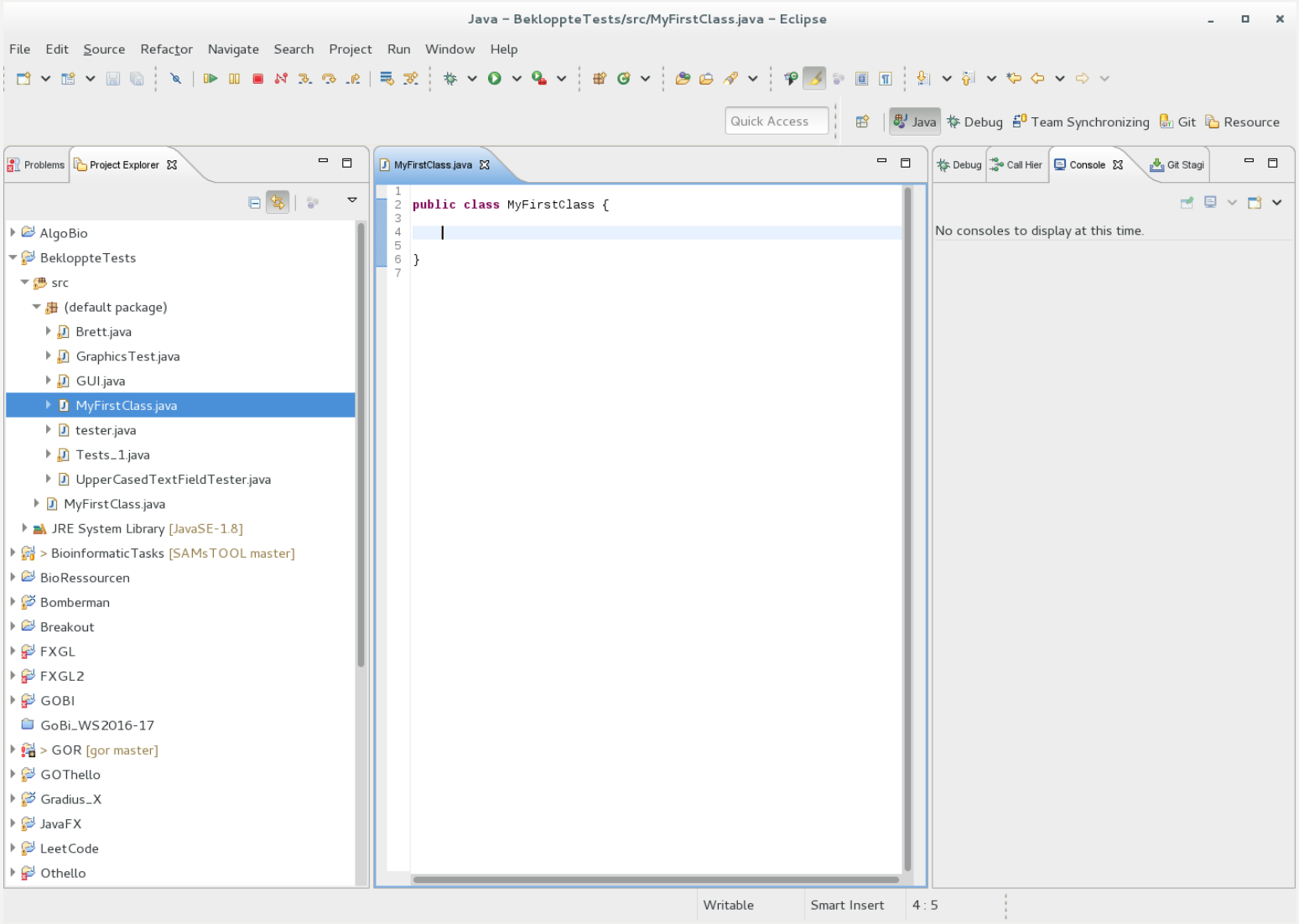
Eclipse


Java – Eclipse
File Edit Navigate Search Project Run Window Help
Quick Access
Java Debug Team Synchronizing Git Resource
Problems Project Explorer
Debug Call Hierarchy Console Git Staging
No consoles to display at this time.
AlgoBio
BekloppteTests
src
New
Go Into
Open Type Hierarchy F4
Show In Umschalt+Alt+W
Copy Strg+C
Copy Qualified Name
Paste Strg+V
Delete
Remove from Context
Build Path
Source Umschalt+Alt+S
Refactor Umschalt+Alt+T
Import...
Export...
Refresh F5
References
Declarations
Run As
Debug As
Validate
Restore from Local History...
Project...
Annotation
Class
Enum
Interface
Package
Example...
Other... Strg+N

Eclipse


New Java Class
Java Class
Create a new Java class.
Source folder:
BekloppteTests/src
Browse...
Package:
(default)
Browse...
Enclosing type:
Browse...
Name:
Modifiers:
public
package
private
protected
abstract
final
static
Superclass:
java.lang.Object
Browse...
Interfaces:
Add...
Remove
Which method stubs would you like to create?
public static void main(String[] args)
Constructors from superclass
Inherited abstract methods
Do you want to add comments? (Configure templates and default value here)
Generate comments
Cancel
Finish

Eclipse


Java – BekloppteTests/src/MyFirstClass.java – Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Java Debug Team Synchronizing Git Resource
Problems Project Explorer
AlgoBio
BekloppteTests
src
(default package)
Brett.java
GraphicsTest.java
GUI.java
MyFirstClass.java
tester.java
Tests_1.java
UpperCasedTextFieldTester.java
MyFirstClass.java
JRE System Library [JavaSE-1.8]
> BioinformaticTasks [SAMsTOOL master]
BioRessourcen
Bomberman
Breakout
FXGL
FXGL2
GOBI
GoBi_WS2016-17
> GOR [gor master]
GOThello
Gradius_X
JavaFX
LeetCode
Othello
MyFirstClass.java
1 public class MyFirstClass {
2
3
4 }
5
6
7
Debug Call Hier Console Git Stagi
No consoles to display at this time.
Writable Smart Insert 4 : 5

Die jShell

- Vereinfacht den Ablauf:

- Write a complete program.
- Compile it and fix any errors.
- Run the program. → Try the code.
- Figure out what is wrong with it.
- Edit it.
- Repeat the process.

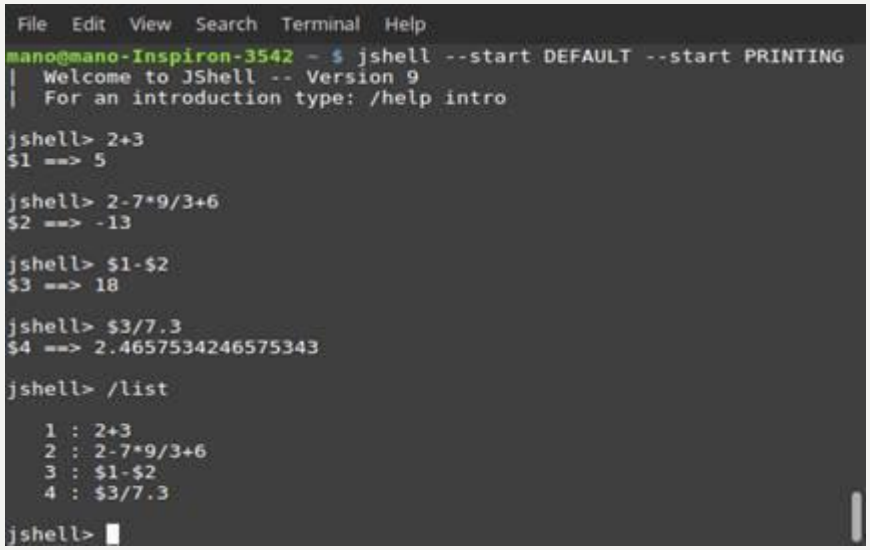
- Wichtige Befehle:

- jshell –v
- /exit
- /help intro
- /list

Die jShell

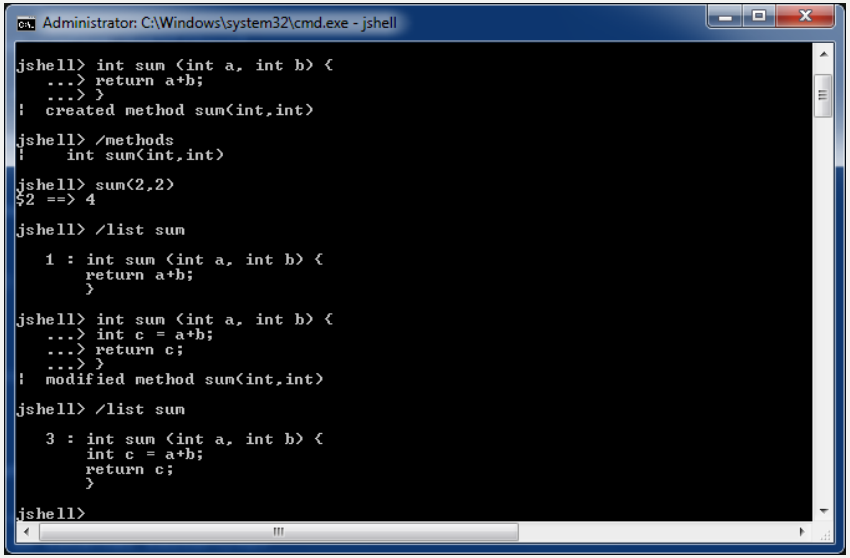
Beispiele für:

Berechnungen



https://www.developer.com/imagesvr_ce/5633/Shell07.jpg

Funktionen



https://cdn2.howtodoinjava.com/wp-content/uploads/2017/08/Working-with-Methods-in-JShell.png

Die Main-Methode

- … ist der Einstiegspunkt in ein Java-Programm

- … stellt eine Verbindung zu allen beteiligten Klassen her

```
1  public class HelloWorld
2  {
3    public static void main(String[] args)
4    {
5      int a = 1;
6      int b = a+1;
7
8      System.out.println("a+b = " + (a+b));
9    }
10 }
```

Parameter *args*

- … nimmt die Eingaben aus der Konsole entgegen

- … alle Argumente werden als String übergeben

- … das erste Argument kann über args[0] angesprochen werden

```
public static void main(String[] args){
 String s = args[0]; //erster Paramter
 System.out.println(s);
 ...
 System.out.println(args[args.length-1]); //letzter
   Parameter
}
```

Datentypen

Datentypen in Java:

Buchstaben:	**char (Character)**
Wörter:	**String**
Zahlen aus N:	**int (Integer)**
Kommazahlen:	**float, double**

https://de.wikibooks.org/wiki/Java_Standard:_Primitive_Datentypen

Umwandeln von Datentypen:

```
String s1 = "5";
String s2 = "10.5";

int f1 = Integer.parseInt(s1);
float f2 = Float.parseFloat(s2);
```

IF(), ELSE IF(), ELSE ()

Programmabfluss kann durch Fallunterscheidungen verändert werden:

```
1 if (Ausdruck){ //   z.B.: a == b, a!=b, a>b s...
2   //Falls der Ausdruck erfuellt ist
3 }
```

```
1 if (Ausdruck){
2   //Falls der Ausdruck erfuellt ist
3 }else{
4   //Falls der Ausdruck nicht erfuellt ist
5 }
```

```
1 if (Ausdruck){
2   //Falls der Ausdruck erfuellt ist
3 }else if (2. Ausdruck){
4     //Falls der Ausdruck nicht erfuellt ist
5 }...{
6 }else{
7     //Falls alle anderen Ausdruecke nicht erfuellt sind
8 }
```

IF(), ELSE IF(), ELSE ()

Der Datentyp String muss mit .equals(anotherString) auf Gleichheit geprüft werden:

```
String s1 = "Tutorium";
String s2 = "Bioinformatik";
String s3 = "Tutorium";

int a = 2;
int b = 3;

if(s1.equals(s2)){
    System.out.println("Wir sind gleich");
}else if(a==b || s1.equals(s3)){
    System.out.println("Mindestens einer von uns ist gleich");
}else{
    System.out.prinln("Keiner ist gleich");
}
```

javac

Zwei einfache Zeilen:

javac Klasse.java

java Klasse Parameter