

Programmering intro

Contents

1	Intro	2
2	Programming languages, compilers, and interpreters	2
3	A sip of Java	2
4	Variables	3
5	Documentation	3
6	Loops	3

1 Intro

The following document are the basic information to start programming in Java. For more code information the exercises can be viewed.

2 Programming languages, compilers, and interpreters

Most languages are high-level languages and therefore easy for humans to read.

High-level languages have to be compiled to machine language.

Assembly language is low level and almost machine language but with some translations.

Interpreter is a live translation from high-level language to machine code, but is way slower than a compiler but more flexible from machine to machine.

Java is a combination of interpreter and compiler. First the code is compiled into bytecode from which a Java virtual machine interprets it. This makes the bytecode portable from machine to machine and light-weight when a compiler is not needed.

JVM uses a Just-in-Time (JIT) compiler, which compiles code segments and save them for future use.

3 A sip of Java

The most simple Java program is:

```
1 public class ProgramName {  
2     public static void main(String[] args) {  
3     }  
4 }
```

Here ProgramName is also the file name, so in this case the file name is *ProgramName.java*

Main is a method and methods are used like functions.

The main method is a static method, which means it does not relate to any object but rather only relates to the class.

Main is a void which means it will not return anything.

The reason for the access modifier *public* is due to otherwise the JVM would not be able to access the main method class, due to it being outside the

application.

To import a library the syntax is *import java.util.Scanner* as an example for the Scanner class from the library java.util

To invoke a method in an object first the object name is written then a dot followed by the method and parenthesis with optional arguments *System.out.println("test")*. Here the object is System.out which itself is part of the System class.

4 Variables

Primitive data types are variables declared with *int, char, double...* unlike class type which is declared in classes

The variable identifier (name) can only contain letters, numbers and underscore and can not start with a number

Identifiers for variables should never be capitalized unlike classes which is capitalized

For constants the keyword *final* can be used in the declaration and these constants often use underscore for spaces and are all capitalized.

5 Documentation

To comment code a single line is done by *//*, multiline is */**/* and for compatibility with javadoc */** */* is used.

A code should always contain a header comment which describes the code, who made it, last edit and purpose like assignment.

6 Loops

The increment should always happen at the end of the loop.

The most simple loops are while which first check statement and then maybe execute the body of the loop, and do while which first executes then checks the statement if a loop should be done.