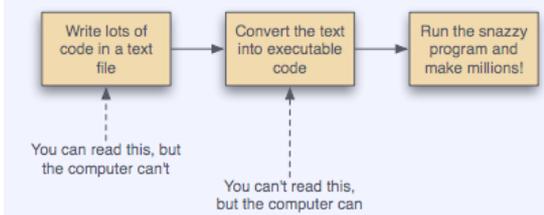


Introduction

A **program** is a set of instructions that can be converted to instructions a computer can understand and thus execute.



JVM is a program which enables java programs to run on many diff computer platforms

Java syntax

- Some words are reserved and cannot be used for variable names
- Variable names cannot begin with numbers
- Blocks of code are delineated with braces { }
- Array items are accessed with []
- Statements should end with ;
- Strings are limited by double quotes “ “

Compile: javac HelloWorId.java

Run: java HelloWorId

Variables/Expressions

Storing and manipulating info in memory

Binary numbers

Boolean values: TRUE = 1 FALSE = 0

Integers are stored as binary

base 10	base 2 / binary	expansion
0	0	0
1	1	1×2^0
2	10	$1 \times 2^1 + 0 \times 2^0$
3	11	$2^1 + 2^0$
5	101	$2^2 + 2^0$

bytes

everything in a computer is stored as bits

byte = 8 bits

bits = binary digit

bits

1's and 0's are easy to store as on/off so binary has become universal way to store electronic info

variables: store info in memory to access it later

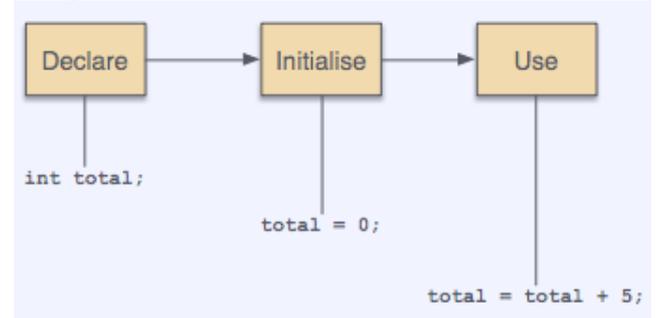
- Can only start with a letter, _ or \$ (not convention to start with _ though)
- No hyphens
- Use camelCase
- Start with lowercase

primitive types

simplest type e.g. int → uses fixed amount of memory and can be changed with simple math operations

objects → customised type of variables e.g. String

object can contain many primitive types as well as many other objects. Amount of memory used by an object can change over time



declare → telling computer that we are storing an integer value

initialise → need to set a value for the variable

blocks of code

collection of statements delineated by {} → instructions get executed in sequence

scope is where the identifier can be used

once it has been declared, a variable will only persist within its scope

the scope of a local variable is essentially the block in which its declared

types of variables

byte, short, int, long

- Numerical values
- Int for integer variables → we don't use int if the range is insufficient or memory is an issue

Float and double

- Floating point types
- Float uses 4 bytes of memory, double uses 8 bytes
- Bits used to describe the sign/exponent/mantissa (fractional part)
- Use double

Floating point numbers

- Don't compare with == as rounding errors affect the comparison
- Instead, we test the difference between expected answer and returned value
- E.g. (number-3.1415926 < 0.000001)

String

- Sequence of characters stringed together
- It is an object not a primitive type

'+' operator creates a new string from the concatenation of 2 existing strings

. length() method returns an int

. equals() method returns a Boolean (don't use ==)

default values

when variables are not initialised they get default values

The default value for the integer types **char**, **int**, **byte**, **long**, **byte** is

0

The default value for a **String** is

null

The default value for a **boolean** (Boolean) variable is

false

Main method

Public static void main(String[] args)

Parsing

Double.parseDouble → String to double

casting

allows programmer to explicitly tell compiler to treat a variable or expression as another type
temporarily change the data type

integer division: $2/5 = 0.0$

double portion = (double) litres / (double) persons
= (double) (litres/persons) = 0.4

break & continue statement

Break is special reserved word

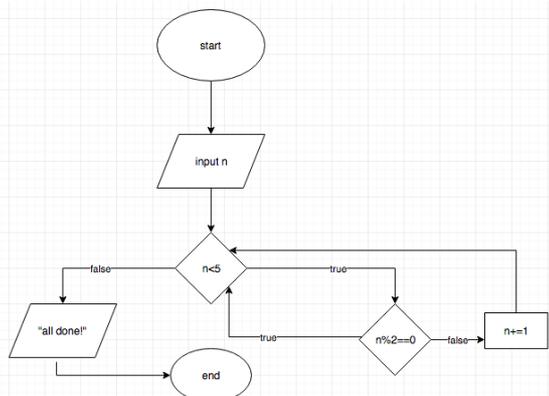
- In loops, it means "break from current iteration of this loop"
- Not allowed in "if" statements

Continue is a special reserved word

- In loops, it means "skip the rest of this iteration and go on to the next one"
- Only allowed in body of a loop

Flowchart

Input/output = Parallelogram
Process/annotation = rectangle
Terminal = flat oval
Decision = diamond
e.g. while with continue:



Arrays

A contiguous block of memory containing multiple values of the same type

- Solves problem of having many variables

Declare

int[] x;

declare and initialise

double [] z = new double[y];

size of array

array.length

ArrayIndexOutOfBoundsException

Attempt to access an index that has no elements

Methods

A function is a series of instructions that produces output based on input

A method is a java term used to describe a function associated with a class or object

- Separate part of a program that performs some operations which can be invoked somewhere else in the program
- Methods can accept arguments and either return a variable or nothing (void)

Why use methods

1. Tidy and easy to understand
2. Allows for code reuse
3. Reduces change of error

A method consists of a method prototype/header & body
<modifiers> <type> <name>(<parameters>) {
<body> }

modifiers = public/private/static

parameters = comma-separated variable declarations

Method signature: name & arguments

Method declaration: return type, name & arguments

Arguments: Sth that is passed to a method

Parameter: Sth used by a method

Returning

- Option of getting sth back from the method (a message or value)

Calling a method

- Get info we want method to handle
- Invoke method by using name and supplying info as arguments
- Do something with returned item
- Return ≠ print
- Once you return the method ceases execution
- Can only return 1 data type

Access modifiers

Local variables / values

Method arguments are new variables that exist for the code block {} are called local variables

Static methods