**Computer and internet basics**

**What is a programming language?**

A programming language is a special language programmers use to develop software programs, scripts, or other sets of instructions for computers to execute.

**What type of programming language is Java?**

Java is a high-level programming language developed by Sun Microsystems. The Java syntax is similar to C++, but is strictly an object-oriented programming language.

**What is a statement?**

A statement in Java forms a complete command to be executed and can include one or more expressions.

**What is a variable?**

A variable is the name given to a memory location. It is the basic unit of storage in a program.

**What is variable declaration?**

* Primitive data types (low first letter) – includes whole numbers (byte, short, int, long), Fractional numbers (float, double, boolean and char)
* Non-primitive data types (big first letter) - such as String, Arrays and Classes

**What is variable assignment?**

Assignment in Java is the process of giving a value to a primitive-type variable or giving an object reference to an object-type variable.

**What kind of variable types do you know?**

* Primitive data types - includes byte, short, int, long, float, double, boolean and char
* Non-primitive data types - such as String, Arrays and Classes

**What type of operators do you know?**

Aritmetic

+ (Addition) - Adds values on either side of the operator.

- (Subtraction) - Subtracts right-hand operand from left-hand operand.

\* (Multiplication) - Multiplies values on either side of the operator.

/ (Division) - Divides left-hand operand by right-hand operand.

% (Modulus) - Divides left-hand operand by right-hand operand and returns remainder.

++ (Increment) Increases the value of operand by 1.

-- (Decrement) Decreases the value of operand by 1.

Relational

== (equal to) - Checks if the values of two operands are equal or not, if yes then condition becomes true.

!= (not equal to) - Checks if the values of two operands are equal or not, if values are not equal then condition becomes true.

> (greater than) - Checks if the value of left operand is greater than the value of right operand, if yes then condition becomes true.

< (less than) - Checks if the value of left operand is less than the value of right operand, if yes then condition becomes true.

>= (greater than or equal to) - Checks if the value of left operand is greater than or equal to the value of right operand, if yes then condition becomes true.

<= (less than or equal to) - Checks if the value of left operand is less than or equal to the value of right operand, if yes then condition becomes true.

Bitwise

& (bitwise and) - Binary AND Operator copies a bit to the result if it exists in both operands.

| (bitwise or) - Binary OR Operator copies a bit if it exists in either operand.

^ (bitwise XOR) - Binary XOR Operator copies the bit if it is set in one operand but not both.

~ (bitwise compliment) - Binary Ones Complement Operator is unary and has the effect of 'flipping' bits.

<< (left shift) - Binary Left Shift Operator. The left operands value is moved left by the number of bits specified by the right operand.

>> (right shift) - Binary Right Shift Operator. The left operands value is moved right by the number of bits specified by the right operand.

>>> (zero fill right shift) - Shift right zero fill operator. The left operands value is moved right by the number of bits specified by the right operand and shifted values are filled up with zeros.

Logical

&& (logical and) - Called Logical AND operator. If both the operands are non-zero, then the condition becomes true.

|| (logical or) - Called Logical OR Operator. If any of the two operands are non-zero, then the condition becomes true.

! (logical not) - Called Logical NOT Operator. Use to reverses the logical state of its operand. If a condition is true then Logical NOT operator will make false.

Assignment

= Simple assignment operator. Assigns values from right side operands to left side operand.

+= Add AND assignment operator. It adds right operand to the left operand and assign the result to left operand.

-= Subtract AND assignment operator. It subtracts right operand from the left operand and assign the result to left operand.

\*= Multiply AND assignment operator. It multiplies right operand with the left operand and assign the result to left operand.

/= Divide AND assignment operator. It divides left operand with the right operand and assign the result to left operand.

%= Modulus AND assignment operator. It takes modulus using two operands and assign the result to left operand.

<<= Left shift AND assignment operator.

>>= Right shift AND assignment operator.

&= Bitwise AND assignment operator.

^= bitwise exclusive OR and assignment operator.

|= bitwise inclusive OR and assignment operator.

Conditional Operator ( ? : )

Conditional operator is also known as the ternary operator. This operator consists of three operands and is used to evaluate Boolean expressions. The goal of the operator is to decide, which value should be assigned to the variable.

Instanceof Operator

This operator is used only for object reference variables. The operator checks whether the object is of a particular type (class type or interface type).

**What is the Hello World! application?**

A "Hello, World!" program generally is a computer program that outputs or displays the message "Hello, World!". Because it is very simple in most programming languages, it is often used to illustrate the basic syntax of a programming language and is often the first program that those learning to code write.

**What is string concatenation?**

Using the + operator is the most common way to concatenate two strings in Java. You can provide either a variable, a number, or a String literal (which is always surrounded by double quotes).

**What is a block?**

A block in Java is a group of one or more statements enclosed in braces. A block begins with an opening brace ({) and ends with a closing brace (}). Between the opening and closing braces, you can code one or more statements.

**What is an if statement?**

An if statement can be followed by an optional else statement, which executes when the Boolean expression is false.

**What is a for-loop statement?**

The for loop loops through a block of code a number of times.

**What is a while-loop statement?**

The while loop loops through a block of code as long as a specified condition is true.

**What is a do-loop statement?**

A do-loop is similar to a while-loop, except that a do-loop is guaranteed to execute at least one time.

**What is a switch statement?**

Use the switch statement to select one of many code blocks to be executed.

**Instructions & memory**

**Memory Address**

A memory address is an exact assigned location in RAM used to track where information is stored.

**Instruction Set**

The instruction set provides commands to the processor, to tell it what it needs to do.

* ADD - Add two numbers together.
* COMPARE - Compare numbers.
* IN - Input information from a device, e.g., keyboard.
* JUMP - Jump to designated RAM address.
* JUMP IF - Conditional statement that jumps to a designated RAM address.
* LOAD - Load information from RAM to the CPU.
* OUT - Output information to device, e.g., monitor.
* STORE - Store information to RAM.

**Register**

A register is a temporary storage area built into a CPU. Some registers are used internally and cannot be accessed outside the processor, while others are user-accessible. Most modern CPU architectures include both types of registers.

**BUS**

Each bus inside a computer consists of set of wires that allow data to be passed back and forth. Most computers have several buses that transmit data to different parts of the machine.

**Input**

Whenever you enter data into your computer, it is referred to as input.

**Output**

Data generated by a computer is referred to as output.

**Code**

**Programming languages**

A programming language is a set of commands, instructions, and other syntax use to create a software program. Languages that programmers use to write code are called "high-level languages." This code can be compiled into a "low-level language," which is recognized directly by the computer hardware.

**Compiled vs. Interpreted**

Interpreters and compilers are similar, since they both recognize and process source code. However, a compiler does not execute the code like and interpreter does. Instead, a compiler simply converts the source code into machine code, which can be run directly by the operating system as an executable program. Interpreters bypass the compilation process and execute the code directly.

**Source code**

Source code, often referred to as simply the "source" of a program, contains variable declarations, instructions, functions, loops, and other statements that tell the program how to function.

**Machine code**

Machine language, or machine code, is a low-level language comprised of binary digits (ones and zeros). High-level languages, such as Swift and C++ must be compiled into machine language before the code is run on a computer.

**Compiler**

A compiler is a software program that compiles program source code files into an executable program. It is included as part of the integrated development environment IDE with most programming software packages.