

Concepts and Basics

3.1

Compiled languages translate the entire program into equivalent machine code before carrying out any statement. Interpreted languages translate the program line by line, alternately reading lines and carrying out commands. Generally, interpreted languages are slower as they need to re-interpret lines in a loop. Compiled code is platform-dependent unlike interpreted code.

3.2

Java is not directly executed on the computer, rather it is executed on the Java Virtual Machine, the programs are run inside a virtual machine sandbox, allowing for development of virus free systems. Java code can be run on multiple platforms, it is a language that you can write once and run anywhere.

3.3

A variable is a named storage location, that stores a value of a particular data type. A constant is a data item which cannot change. They have different naming conventions in java, variables are a mix of upper and lower case whereas constants are uppercase separated by an underscore. Unlike variables once constants have been created its value cannot be altered in any way. Both are a way of assigning a value to memory, which can be used later in the program.

3.4

The 8 primitive data types are boolean, char, byte, short, int, long, float and double. The difference between byte, short, int and long is the range the value stored has as they have different sizes despite all containing signed integers. Float and double both hold floating points; a float is 32 bits whereas a double is 64. Boolean contains a true or false value and is the only non-numerical primitive data type. Finally, char stores a Unicode character and is of size 16 bits.

```
int a= 012;  
byte b= 120;  
short c=23461;  
double d=1e-2;  
float e=1.02f;  
char f= "B";  
boolean g = true;  
long h =231201;
```

3.5

Casting is the process of converting one numeral type to another numeral type. Implicit casting is where you increase the memory space allocated (also known as widening casting), conversely explicit casting is the process of reducing the memory space allocated to the variable. Java tries to be secure so you must explicitly state the type when trying to reduce the amount of space allocated.

3.6

Overflow is when the number assigned is too big to be stored in the primitive types that have been allocated. This is usually due to operations with integer types.

e.g.

byte d=130; //byte can only store integers up to 127

3.7

The four main features of object oriented programming are encapsulation, abstraction, inheritance and polymorphism. Encapsulation is when each object maintains a private state inside a class, other objects cannot directly access this state, they can only use public functions. Abstraction is an extension of encapsulation; it is selecting data from a larger set to only show the relevant details to the object. It is the process of fetching, removing or selecting user information from a larger pool. Inheritance is the ability of a property to acquire some or all properties of another object. Polymorphism allows us a way to use a class in the same way as its parent so there is no confusion with mixing types.