Concepts answers

3.1) An interpreted language will use an interpreter to translate the program line by line, carrying out the statements as it does this. This makes it much slower at translating the program. In contrast, a compiled programming language translates the entire program into equivalent machine codes or an executable program before carrying out any of the statements. These are faster than interpreted languages since it can optimize the code.

3.2) Java is secure and platform-independent since Java will use a compiler (JAVAC) to generate a byte code that will be executed on the Java virtual machine (JVM) allowing it to run on the platform. Since it is running in a controlled environment, it is secure.

3.3) The similarities between a constant and variable is that they both refer to a named storage location that stores a value of a data type. However, a constant cannot be changed after declaration while variables can be. They also have various naming conventions with variables being mixed case, the first letter lower case and constants being all uppercase with each word separated by an underscore.

3.4) The eight primitive data types are byte, short, int, long, float, double, boolean and char. The similarities between byte, short, int and long are that they store integer numbers, but they are different sizes (for example byte can only store up to 1 byte and short can store 2 bytes). The similarities between float and double are that they store fractional numbers, but like with integer literals, they are different sizes. The char primitive data type is a numerical type like integer literals, so arithmetic operations are allowed, unlike integer literals stores a single character. A boolean is different from all the other types, not numeric and only having a 1-bit size. It also only has two possible values true or false.

```
byte num;
num = 10;

short num;
num = 128;

int num;
num = -100;

long num;
num = 1000000000;

float height;
height = 0.2f;

double height;
height = 0.0000000354f;

boolean valid;
valid = true;

char letter;
letter = ‘t’;
```


Name: Jack Shaw
Student Number: 700018370

3.5) Casting is when you assign a value of a data type into a different data type. Implicit casting is done automatically, but explicit casting must be done manually with the type casting to being specified.

3.6) An overflow is when the absolute value is too large for the data type it is being stored in. This results in incorrect behavior as the number will overflow back to the smallest number store in this data type but there will be no error message. An example of an overflow is:

```
byte num;
num = 127;
System.out.println(num);
num++;
System.out.println(num);
```

This will print out:
127
-128 (overflow, absolute is 128)

3.7) The four main features of object-oriented programming are encapsulation, abstraction, inheritance, polymorphism. Encapsulation is the mechanism of wrapping the attributes and the methods together into a single unit. Also, called data hiding as the attributes are hidden from other classes. This can be achieved through the usage of private attributes. These can be set and got through setter and getter methods. These methods act as an interface between this class and the others. Abstraction is the hiding of irrelevant data to reduce the complexity of a program. This can be done with encapsulation although there are other ways. Inheritance is when classes can be derived from another class. The child class will inherit from the parent class any methods and fields. Polymorphism is the ability to perform a single action in different ways. This can be done with overloading and overriding methods.