

CS2500 Autumn 2011 Solutions

Question 1 (a)

```
public static void main (String[] args) {
    for (String argument : args ) {
        System.out.println( argument );
    }
}
```

Question 1 (b)

```
private void printWord () {
    Scanner scan = new Scanner(System.in);
    String currentWord = null;
    boolean stopWord = false;

    while(!(stopWord) && scan.hasNext()) {
        currentWord = scan.next();
        if (currentWord.equals("stop") {
            scan.close;
            stopWord = true;
        }
        System.out.println(currentWord);
    }
}
```

Question 2 (a)

An object is an instance of a given class. The class is the blueprint for an object, it describes the objects instance variables, instance methods and how the object should be constructed. The behaviour of an object can be interpreted and also the state of the object.

Question 2 (b)

Object references in java are variables that hold information on an object. Objects do not have names, just types and locations in memory (and fields and methods of course). When an object is required the objects reference is called.

Example

```
Student studentName = new Student ("Joe");
```

`studentName` an object reference in the above example.

Question 2 (c)

Visibility Modifiers allow the programmer to hide information by classifying what they want to be seen and not be seen. It allows for clear abstraction of program components and provides security. Java by default provides limited visibility but to ensure a fully secure program the programmer must classify code by using the following modifiers:

- private
- protected
- no modifier (default)
- public

The above visibility modifiers allow code to be seen/changed in various ways, as illustrated in the table below.

Modifier	Class	Package	Subclass	World
public	Yes	Yes	Yes	Yes
protected	Yes	Yes	Yes	No
no modifier	Yes	Yes	No	No
private	Yes	No	No	No

Question 3 (a)

Primitive Types (in order):

- char
- byte
- short
- int
- long
- float
- double
- boolean

Question 3 (b)

Primitive types contain only one value and cannot be broken down any further. An object is a data type that a class programmer created, such as a String. Objects come with instance variables and methods that can alter those instance variables or return information about the object (such as the `.length` method for String).

Primitives only store one value whereas objects can store not only more than one value but more than one different data type including other objects.

Question 3 (c)

Java is an object orientated language where everything is an object; primitive types go against this principle so require a Wrapper Class to be included in this. All primitive data types in Java have a corresponding Wrapper Class and this encapsulates the single value for the primitive type.

The purpose of Wrapper Classes is so that the primitives can be included in activities reserved for objects, like added to collections and also provide an assortment of utility functions for primitives.

Example

Primitive Type	Wrapper Class
Int	Integer
Char	Character
boolean	Boolean
Double	Double