Assignment 2

1. Why we need packages in java?

A package in Java is used to group related classes. Think of it as a folder in a file directory. We use packages to avoid name conflicts, and to write a better maintainable code.

2. What is the default imported package?

Java compiler imports java. lang package internally by default.

3. What is Class? What is Object?

Class are a blueprint or a set of instructions to build a specific type of object.
Object is an instance of a class. An object in OOPS is nothing but a self-contained component which consists of methods and properties to make a particular type of data useful.

4. Why we need constructor?

We use constructors to initialize the object with the default or initial state. The default values for primitives may not be what are you looking for.
Another reason to use constructor is that it informs about dependencies. In other words, using the constructor, we can request the user of that class for required dependencies.

5. What is the default value of local variable? What is the default value of instance variable?

The local variables do not have any default values in Java. This means that they can be declared and assigned a value before the variables are used for the first time, otherwise, the compiler throws an error.
Instance variables have default values. For numbers, the default value is 0, for Booleans it is false, and for object references it is null. Values can be assigned during the declaration or within the constructor.

6. What is garbage collection?

Java garbage collection is the process by which Java programs perform automatic memory management.

7. The protected data can be accessed by subclasses or same package. True or false?

True.

8. What is immutable class?

Immutable class in java means that once an object is created, we cannot change its content

9. What's the difference between "==" and equals method?

== checks if both objects point to the same memory location whereas . equals() evaluates to the comparison of values in the objects

10. What is wrapper class?

A Wrapper class is a class whose object wraps or contains primitive data types. When we create an object to a wrapper class, it contains a field and in this field, we can store primitive data types.

11. What is autoboxing?

The conversion of a primitive value into an object of the corresponding wrapper class is called autoboxing.

12. StringBuilder is threadsafe but slower than StringBuffer, true or false?

False.

13. Constructor can be inherited, true or false?

False.

14. How to call a super class's constructor?

Using keyword "super".

15. Which class is the super class of all classes?

Object class.

16. Create a program to count how many files/folders are there inside one folder.
    - the count method should take a parameter called Criteria like this: count(Criteria criteria){}
    - For Criteria class, multiple conditions should be included such as: folder path, includeSubFolder or not, the extension of the file be counted and so on.
    - Optional: Take the input from keyboard.
    - Take care of the invalid inputs. Exception handling.
    - Get proper result displayed.
      "There are XXX file(s) and XXX folder(s) inside folder XXX with extension XXX." or something user friendly.

```
package file;
import java.io.File;
import java.util.ArrayList;

public class FileCounter {
   ArrayList<File> fileList;
   File root;

   public FileCounter(Criteria criteria) {
      root = new File(criteria.path);
      fileList = new ArrayList<>();
   }

   public void searchFiles() {
      File[] files = root.listFiles();
      int length = files.length;
      for (int i = 0; i < length; i++) {
         if (files[i].isDirectory()) {
            root = files[i];
            searchFiles();
         } else {
            fileList.add(files[i]);
         }
      }
   }

   public void countFiles() {
      int totalSize = 0;
```

```
        System.out.println("文件数:" + fileList.size());
        for (int i = 0; i < fileList.size(); i++) {
            totalSize += fileList.get(i).length();
        }
        System.out.println("There are "+totalSize+ " files and "+ fileList.size()+" files");
    }


    public static void main(String[] args) {
        FileCounter counter = new FileCounter(criteria);
        counter.searchFiles();
        counter.countFiles();
    }

}
```