Assignment 1

1. What is JDK? JRE? JVM?

The Java Development Kit (JDK) is one of three core technology packages used in Java programming, along with the JVM (Java Virtual Machine) and the JRE (Java Runtime Environment). It's important to differentiate between these three technologies, as well as understanding how they're connected:
The JVM is the Java platform component that executes programs.
The JRE is the on-disk part of Java that creates the JVM.
The JDK allows developers to create Java programs that can be executed and run by the JVM and JRE.

2. What is java compiler?

A Java compiler is a compiler for the programming language Java. The most common form of output from a Java compiler is Java class files containing platform-neutral Java bytecode, but there are also compilers that output optimized native machine code for a particular hardware/operating system combination, most notably the now discontinued GNU Compiler for Java.
Most Java-to-bytecode compilers do virtually no optimization, leaving this until run time to be done by the Java virtual machine (JVM).
The JVM loads the class files and either interprets the bytecode or just-in-time compiles it to machine code and then possibly optimizes it using dynamic compilation.
A standard on how to interact with Java compilers programmatically was specified in JSR 199.

3. Why is java platform independent?

Java is a platform-independent language. In other words, you can write your code once and then run it anywhere, on any platform that provides the environment to run it. This environment is the Java Virtual Machine (JVM). The JVM should be present to execute the code. The JVM is different for each platform. In the case of Java, platform independence does not mean that you can run the code anywhere; you can run it wherever the environment is provided. This is the key point of platform independence in Java.

4. What is IDE? Why is it important for developers?

Integrated development environments (IDE) are applications that facilitates the development of other applications. Designed to encompass all programming tasks in one application, one of the main benefits of an IDE is that they offer a central interface with all the tools a developer needs, including:

Code editor: Designed for writing and editing source code, these editors are distinguished from text editors because work to either simplify or enhance the process of writing and editing of code for developers
Compiler: Compilers transform source code that is written in a human readable/writable language in a form that computers can execute.

Debugger: Debuggers are used during testing and can help developers debug their application programs.
Build automation tools: These can help automate developer tasks that are more common to save time.

5. Is java case sensitive?

Java is a case-sensitive language, which means that the upper or lower case of letters in your Java programs matter.

6. What do the following key words do?

static,The static keyword in Java is used to share the same variable or method of a given class.
final, The final keyword is used in several contexts to define an entity that can only be assigned once.
public, The public keyword is an access modifier used for classes, attributes, methods and constructors, making them accessible by any other class.
private, The private keyword is an access modifier used for attributes, methods and constructors, making them only accessible within the declared class.
void, The void keyword specifies that a method should not have a return value.
null, In Java programming, null can be assigned to any variable of a reference type (that is, a non-primitive type) to indicate that the variable does not refer to any object or array.
package, A package is a namespace that organizes a set of related classes and interfaces.
class, A class — in the context of Java — is a template used to create objects and to define object data types and methods.
new, The new operator is used in Java to create new objects. It can also be used to create an array object.

7. What is primitive type and reference type?

Variables in Java are classified into primitive and reference variables. From the programmer's perspective, a primitive variable's information is stored as the value of that variable, whereas a reference variable holds a reference to information related to that variable.

8. Is parameter passed by value or reference?

Passed by value.

9. What is the output: System.out.println(1 > 0 : “A”:”B”);

No output, will throw error messages.

10. How to define constants in java?

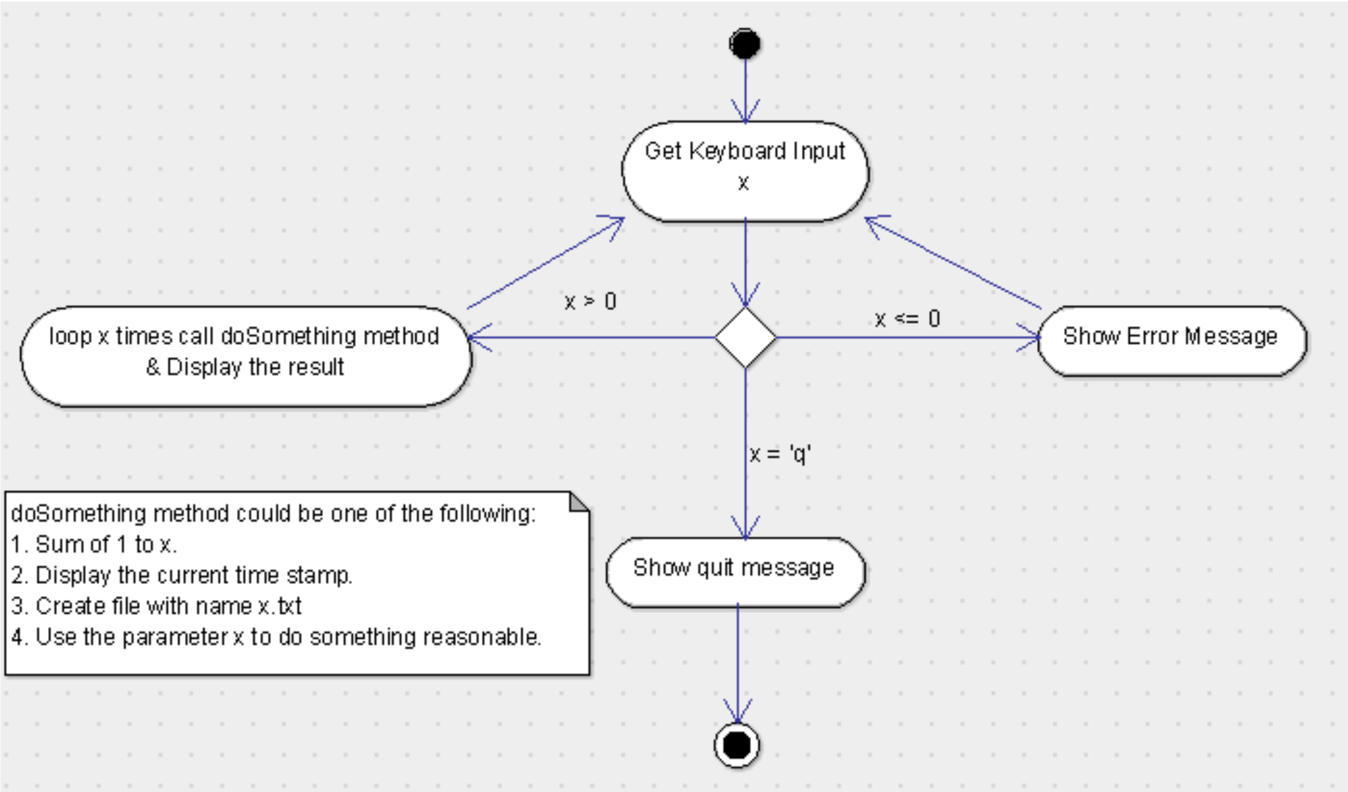
To define a variable as a constant, we just need to add the keyword “final” in front of the variable declaration.

11. What is String? Is it primitive type?

Strings, which are widely used in Java programming, are a sequence of characters. In the Java programming language, strings are objects. The Java platform provides the String class to create and manipulate strings. It’s not primitive type.

12. How to check if a String is representing a number?
Perhaps the easiest and the most reliable way to check whether a String is numeric or not is by parsing it using Java's built-in methods:
Integer. parseInt(String)
Float. parseFloat(String)
Double. parseDouble(String)
Long. parseLong(String)
new BigInteger(String)

13. Write a program to implement the following activity diagram:



doSomething method could be one of the following:
1. Sum of 1 to x.
2. Display the current time stamp.
3. Create file with name x.txt
4. Use the parameter x to do something reasonable.

14. Write a program to merge two array of int.
15. Write a program to find the second largest number inside an array of int.