HW1

1. What is the difference between JDK, JRE, and JVM?

   JVM is Java virtual machine, executes bytecode, provide platform independence

   JRE is Java running environment, includes JVM and standard libraries

   JDK is Java development kit, include JRE and other development tools(compiler, debugger, etc.)

2. What are the main differences between primitive types and reference types?

   primitive types variable store the data value in stack, while reference types variable store the reference (address of the object, and the object is stored in heap)

3. Is Java pass-by-value or pass-by-reference? Explain.

   Java is pass-by-value. Because reference type varaible store the the data address in heap, when it’s passed to function, it copy the address value and pass it to function, and therefore when you access and modify it in the function, it does the modification in the same memory address in heap, which means the original object will also change.

4. Why are Strings immutable in Java?

   Because string is widely used and is repeated frequently. When a string is initialized, Java create a string object and store tit in string constant pool, then next if someone create a same string, there is no need to create a same object, it just give the same reference to the new string variable. Due to this mechanism, strings are immutable cause there might be two variable refer to a same string.

5. What is the String Constant Pool and how does it work?

   String Constant Pool is a special space in heap. When a string is initialized, Java first check if the string is already created in String Constant Pool, if not, it create the string object in string constant pool; if there already exists the same string in String constant pool, if directly return the reference and no need to create a new string.

6. What is the purpose of the final keyword in Java?

to ensure resources immutable.

7. What does the static keyword mean for variables or methods?

   for class variables, means this variable is not bond with a specific instance, it belong to the whole class.

   for class methods, means this method is not bond with a specific instance, it belong to the whole class. you can call it directly through the class name without exactly creating an instance first.

8. What is a static block and when does it run?

   static block executes only one-time when class is first loaded. it runs after the static variables are intialized by default.

9. Can a static method access non-static variables? Why or why not?

   No. Because static method is not bond with a specific instance, non-static variables belong to their different instances.

10. Describe the JVM loading order: static block, static variables, and constructor.

    static variables loading(initialized by default) → static block running (by coding order) → constructor (when an instance is created)

11. What is the difference between a static variable and a constant defined as public static final?

    static variable: mutable, can only be accessed via the class

    public static final: immutable, compile-time constant, can be accessed from anywhere

12. Explain why immutable objects (like String) are thread-safe by design. Provide an example scenario.

    it’s immutable so there is no race condition.

    when multiple thread access a same string, since they cannot modify it, there is no race condition and is thread-safe.

13. What problem does making a class final solve in terms of immutability and inheritance security?

    it prevents subclassing, ensures subclasses cannot introduce mutability (which would break the immutability guarantee) and override behavior.