Assignments

Friday, August 24, 2018 6:31 AM

1. Enlist differences between final, finally, and finalize.
   a. final - makes a variable constant. Cannot change its data.
   b. finally - written after a try/catch block. JVM makes sure it happens whether there's an exception or not. If there is an exception and a program crashes, the JVM makes sure the finally block happens before the program crashes.
   c. finalize - garbage collection. Happens right before garbage collection. Purpose is to close your resources such as database, open files, connections, etc.
2. Exception propagation - checked and unchecked exceptions.
   a. Methods are executed as a stack. When an exception occurs in one method and is not handled, it goes down the stack until it's handled.
   b. Checked exceptions have to be handled or the program won't compile. It's a compile time error.
   c. Unchecked exceptions don't have to be handled. The program will compile and run. But if it's not handled, there will be a runtime error and the program will crash.
3. Demonstrate custom exception.
   a.
```
public class NoLegsException extends Exception {
    public NoLegsException() {
        super("Has no legs. Cannot walk.");
    }
}
```
   b.
```
public void walk() throws Exception {
    throw new NoLegsException();
}
```
4. Exception handling rules in method overriding.

Parent	Child
No declared exception in method	• Cannot declare checked exception • Can declare unchecked exception
Declared exception in	• Can declare the same or a subclass (exception)

method	of the parent's declared exception • Cannot declare a parent (exception) of the parent's declared exception • Can declare no exception

5. Exception classification.
    a. checked exceptions - Compile time error if not handled (ex. IOException)
    b. unchecked exceptions - Runtime error if not handled (ex. ArrayIndexOutOfBoundsException)