HOMEWORK 8
HANDLING JAVA EXCEPTIONS

Qn	Ans	Explanation
1	D	The code does not compile for another reason. A try block is using with a catch or finally block, or both.
2	B	The correct answer is "try, catch, finally". They must be in this order that they would be used together.
3	D	Throwable class inherits from Object. Exception and Error are extended from Throwable. RuntimeException is extended from Exception.
4	A	Error is not caught in Java applications. An Error fails JVM which cannot be recovered.
5	D	The code does not compile because score is just defined within the try block, not within catch and finally blocks.
6	B	While IOException is a checked exception, others are unchecked exceptions.
7	A	The throws keyword is used in method declarations, while the throw keyword is used to throw an exception to the surrounding process.
8	B	If a try statement has catch blocks for both Exception and IOException, The catch block for IOException must appear before the catch block for Exception.
9	D	The answer is "None of the above" because of the throw t code snippet.
10	C	The code does not compile because of line p3. The main() method has to have a try catch statement around line p3 that properly handles the checked exception, or the main() method has to be updated to declare a compatible checked exception.
11	B	While Exception must be handled or declared by the method in which they are thrown, others are unchecked exceptions.
12	A	The code compiles and outputs "1345". Try block throws a ClassCastException and first catch block is skipped. Second catch block is executed. Then, finally block is executed and returns to the main() method with no exception being thrown.
13	C	The finally statement requires brackets{} so option C is the correct answer.
14	C	The code does not compile because catch blocks are used in the wrong order. Because IOException is a superclass of FileNotFoundException, the FileNotFoundException is considered unreachable code.
15	C	"catch" or "finally" or both are required with a try statement.
16	B	An application that throws an exception can choose to handle the exception and avoid termination.
17	D	The code does not compile for another reason. The catch block uses brackets() instead of parentheses{}.
18	B	public in printData() throws Exception would not be allowed in a class implementing the Printer interface. Overridden methods cannot throw new or broader checked exceptions than the one they inherit. Since Exception is a broader checked exception than PrintException.
19	D	None of the above is required to be declared in order to use the Exception, RuntimeException, and Throwable classes in an application because they are already included by default.
20	C	The code does not compile because of line g3 because the catch block is missing a variable type and name.
21	B	A program must handle or declare checked exceptions but should never handle java.lang.Error.
22	B	The code does not compile because of line q2. There is no try catch around line q2 and the method does not declare a compatible checked exception, only an unchecked exception.
23	A	If an exception matches two or more catch blocks, the first one that matches is executed.
24	C	The code does not compile because of the call to compute() in the main() method. The compute() method just throws an unchecked exception but it must be thrown an checked exception.
25	D	The answer is "none of the above". All options are possible but to decide the exception, we must see that code line.
26	B	A StackOverflowError occurs when a program recurses too deeply into an infinite loop, while a(n) NullPointerException occurs when a reference to a nonsexist object is acted upon.
27	C	Option C is "To ensure that exceptions never cause the application to terminate". This is not a reason to add checked exceptions to a method signature.
28	D	The code does not compile because of the order problem of the catch and finally blocks.
29	A	A try statement has zero or one finally block(s) and zero or more catch blocks.
30	D	The code compiles but throws an exception at runtime. Count is assigned to 0. The getDuckies() method ends up computing 5/0 and it leads to an unchecked ArithmeticException at runtime.

31	B	If both the catch block and the finally block each throw an exception, the exception from the finally block.
32	A	The code does not compile because of line m1 because of the roar() method in the BigCat class uses throw instead of throws.
33	A	The answer is "ClassCastException". The question is about casting. The assignment on the second line throws a ClassCastException at runtime.
34	C	Throwable classes will handle all types in a catch block.
35	B	The answer is "II only". If both values are valid non-null String objects, no exception will be thrown, with the statement in the finally block being executed first, before returning control to main() method; therefore, the second statement is a possible output.
36	A	If a try statement has catch blocks for both ClassCastException and RuntimeException, the catch block for ClassCastException must appear before the catch block for RuntimeException.
37	C	"A caller passes invalid data to a method" is the best scenario to use an exception. IllegalArgumentException can be used to alert a caller of missing or invalid data.
38	C	The code does not compile because of an invalid override of the operate() method.
39	D	NullPointerException can be handled with a try-catch block or ignored altogether by the surrounding method.
40	D	The code does not compile because there is no new keyword in the RuntimeException(String) statement's zipper() method. The new keyword is necessary for creating the object being thrown.
41	C	RuntimeException will be printed in the stack trace at runtime. The application exits and the caller see the RuntimeException in the stack trace.
42	D	First three options are each invalid overrides of the method because the return type must be covariant with void. Therefore, answer is "None of the above".
43	D	The code does not compile because the catch block is missing a variable name such as catch (Error e).
44	D	The code compiles, but a stack trace is printed at runtime. Exceptions are handled by the fact that the method declares the checked Exception class in the method signature, which all the exceptions within the class inherit. For this reason, the openDrawbridge() method compiles without issue. The main() method declares Exception in its signature.
45	C	If a try statement has catch blocks for both IllegalArgumentException and ClassCastException, the catch blocks for these two exception types can be declared in any order.
46	D	The code does not compile because the Problem class does not compile. The class RuntimeException is not an interface and it cannot be implemented.
47	D	The code does not compile because of the wrong keyword. Throws key word is invalid so it must be "throw" to compile the code.
48	D	Given an application that hosts a website, "The application runs out of memory" would most likely result in a java.lang.Error being thrown. It is unrecoverable in Java.
49	C	The code does not compile because of line z1. In that line, variable "e" is declared two times so it does not compile.
50	B	The code does not compile because of line x1. The finally block of the snore() method throws a new checked exception on line x1, but there is no try catch block around it to handle it, nor does the snore() method declare any checked exceptions.

Reference

Scott S, Jeanne B.: OCA/OCP Java SE 8 Programmer Practice Tests. Indiana, USA: 2017.