Q1.A1. Access Modifiers define the scope and accessibility of the class, interface, variables etc..

            The four types of access modifiers are

            Public: is used to access anything in a class from anywhere.

            Private: only the same class can access it.

            Protected: same class + different classes in the same package + child class in different packages can access it.

            Default: same class + different classes in the same package can access it.

Q2.A2. Error: The error indicates a problem that mainly occurs due to the lack of system resources and

our application should not catch these types of problems. Some of the examples of errors are

system crash error and out of memory error. Errors mostly occur at runtime that's they belong to an unchecked

type.

            Example of Error:

public class ErrorExample {

   public static void main(String[] args){

      recursiveMethod(10)

   }

   public static void recursiveMethod(int i){

      while(i!=0){

         i=i+1;

         recursiveMethod(i);

      }

   }

}

            Output: Exception in thread "main" java.lang.StackOverflowError

                           at ErrorExample.ErrorExample(Main.java:42)

            Exceptions: are the problems which can occur at runtime and compile time. It mainly occurs in the code

written by the developers.  Exceptions are divided into two categories such as checked exceptions and

unchecked exception.

            Example of Exception

public class ExceptionExample {

   public static void main(String[] args){

      int x = 100;

      int y = 0;

      int z = x / y;

   }

}

             Output: java.lang.ArithmeticException: / by zero

                           at ExceptionExample.main(ExceptionExample.java:7)

Q3.A3.

|  |  |
| --- | --- |
| Checked Exception | Unchecked Exception |
| Checked exceptions occur at compile time. | Unchecked exceptions occur at runtime. |
| The compiler checks a checked exception. | The compiler does not check these types of exceptions. |
| These types of exceptions can be handled at the time of compilation. | These types of exceptions cannot be a catch or handle at the time of compilation, because they get generated by the mistakes in the program. |
| They are the sub-class of the exception class. | They are runtime exceptions and hence are not a part of the Exception class. |
| Here, the JVM needs the exception to catch and handle. | Here, the JVM does not require the exception to catch and handle. |
| Examples of Checked exceptions:   * File Not Found Exception * No Such Field Exception * Interrupted Exception * No Such Method Exception * Class Not Found Exception | Examples of Unchecked Exceptions:   * No Such Element Exception * Undeclared Throwable Exception * Empty Stack Exception * Arithmetic Exception * Null Pointer Exception * Array Index Out of Bounds Exception * Security Exception |