寫一個判斷三角形的程式，必須用到 assert, exception, checkstyle, findbugs 等技巧

```
package app;
import java.util.Arrays;
import java.util.Scanner;
/**
* @author Alvin
*/

public class App {

  /**
  * This is main function of whole project
  * @param args not use
  * @return noting
  */
  public static void main(String[] args) throws Exception {

    Scanner scanner = new Scanner(System.in);
    System.out.println("Input triangle edges:");
    int a = scanner.nextInt();
    int b = scanner.nextInt();
    int c = scanner.nextInt();
    try {
      getTriangleType(a, b, c);
    } catch (Exception e) {
      System.out.println(e);
    }
    scanner.close();
  }
  /**
   *
   * @param a a edge of triangle
   * @param b a edge of triangle
   * @param c a edge of triangle
   * @throws Exception when a,b or c is zero or negative
   */
  public static void getTriangleType(int a, int b, int c) throws Exception {
    assert a > 0 && b > 0 && c > 0 : "one of the edge is negative or zero";
    int[] edges = new int[] { a, b, c };
    //sort the edges
    Arrays.sort(edges);
    // the sum of two shortest edges is little than the biggest edge cannot form a
triangle
    if (edges[0] + edges[1] <= edges[2]) {
      throw new Exception("this three edges cannot form a triangle");
    }
    System.out.println("is triangle");
    if (edges[0] == edges[1] || edges[1] == edges[2]) { // check  there are two
edges is equal
      System.out.println("is isosceles triangle");
    }
```

```
    if (edges[0] == edges[1] && edges[1] == edges[2]) { // check three edges is
equal
      System.out.println("is equailateral triangle");
    }
    //check a square + b square == c square
    if (edges[0] * edges[0] + edges[1] * edges[1] == edges[2] * edges[2]) {
      System.out.println("is right angle triangle");
    }
  }
}
```

心得：利用 assert 去制定規則對於開發時期的開發及測試非常重要，可以防止一些重大的錯誤發生並給予明確的錯誤信息；findbugs 則可以幫助工程師避免一些細微的錯誤；checkstyle 對於程式碼的整潔度及可理解度都有明顯的提升。