App.java

```
package app;

import java.util.Arrays;

public class App {
    public static void main(String[] args) throws Exception {
        System.out.println("Hello Java");
    }

    public static String 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");
        }
        if (edges[0] == edges[1] && edges[1] == edges[2]) { // check three edges is
equal
            return "is equailateral triangle";
        }
        else if (edges[0] == edges[1] || edges[1] == edges[2]) { // check there are
two edges is equal
            if (edges[0] * edges[0] + edges[1] * edges[1] == edges[2] * edges[2]) {
                return "is isosceles right triangle";
            }
            return "is isosceles triangle";
        }
        // check a square + b square == c square
        else if (edges[0] * edges[0] + edges[1] * edges[1] == edges[2] * edges[2])
{
            return "is right angle triangle";
        }
        else {
            return "is triangle";
        }

    }

}
```

AppTest.java

```
package app;

import static org.junit.jupiter.api.Assertions.assertEquals;
import static org.junit.jupiter.api.Assertions.assertThrows;

import org.junit.jupiter.api.AfterAll;
```

```java
import org.junit.jupiter.api.AfterEach;
import org.junit.jupiter.api.BeforeAll;
import org.junit.jupiter.api.BeforeEach;
import org.junit.jupiter.api.Test;

public class AppTest {
    App app = new App();

    // @BeforeAll
    // void beforeAll() {
    //     System.out.println(">>>>>>>>>>>>>>>>>>>>>>>>.");
    // }

    // @AfterAll
    // void afterAll() {
    //     System.out.println("<<<<<<<<<<<<<<<<<<<<");
    // }

    // @BeforeEach
    // void before() {
    //     System.out.println("Test started.");
    // }

    // @AfterEach
    // void after() {
    //     System.out.println("Test finished.");
    // }

    @Test
    void getTriangleTypeTest() {
        try {
            assertEquals("is triangle", App.getTriangleType(2, 3, 4));
            // assertEquals("is equailateral triangle", App.getTriangleType(3, 3,
3));
            // assertEquals("is isosceles triangle", App.getTriangleType(2, 2, 1));
            // assertEquals("is right angle triangle", App.getTriangleType(3, 4,
5));
            // assertEquals("is triangle", App.getTriangleType(3, 5, 6));
            // assertThrows(Exception.class, ()->App.getTriangleType(1, 2, 3));
        } catch (Exception e) {
            e.printStackTrace();
        }
        // assertEquals(true,true);

    }

}
```

在開發的階段利用 test case 去幫助 developer 測試程式碼的正確性，不像平常寫作業，用已設計好的 test case 一次測試多種輸入不要 developer 自己手動輸入，加快了除錯的時間，十分有用及有效率。