Name: Kevin Tran

PID: A16809193

```
public class HandlerTests {
  @Test <-- called an "annotation"
  public void handleRequest1() throws Exception {
    DoubleHandler h = new DoubleHandler();
    String url = "http://localhost:8080/double?num=1";
    URI input = new URI(url);
    String expected = "Doubling 1 produces 2.";
    assertEquals(expected, h.handleRequest(input));
  }
  @Test
  public void handleRequest2() throws Exception {
    DoubleHandler h = new DoubleHandler();
    String url = "http://localhost:8080/double?num=2";
    URI input = new URI(url);
    String expected = "Doubling 2 produces 4.";
    assertEquals(expected, h.handleRequest(input));
  }
}
```

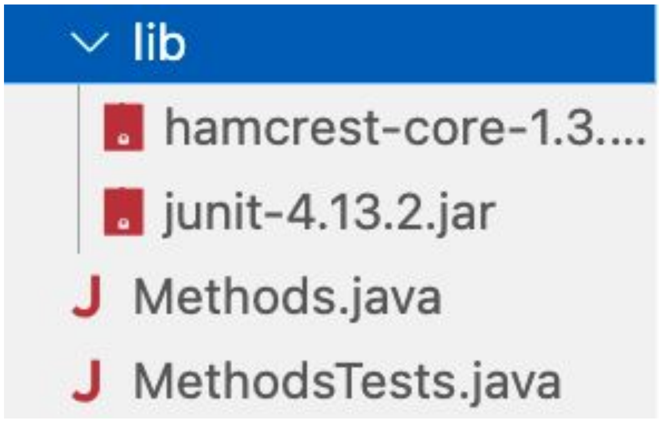
```
class DoubleHandler implements URLHandler {
  String template = "Doubling %d produces %d.";
  public String handleRequest(URI url) {
    if (url.getPath().equals("/double")) {
      String[] params = url.getQuery().split("=");
      if (params[0].equals("num")) {
        int num = Integer.parseInt(params[1]);
        int doubl = num + 2;
        return String.format(template, num, doubl);
      }
    }
    return "404 Not Found!";
  }
}
```

-cp = class path

*.java = all the java files in the directory

```
local$ javac -cp .:lib/hamcrest-core-1.3.jar:lib/junit-4.13.2.jar *.java
local$ java -cp .:lib/hamcrest-core-1.3.jar:lib/junit-4.13.2.jar org.junit.runner.JUnitCore HandlerTests
~~g.junit.runner.JUnitCore HandlerTests~~
JUnit version 4.13.2
.E. <-- . (dot) means test started, E means a test errored
Time: 0.012
There was 1 failure:
1) handleRequest1(HandlerTests)
org.junit.ComparisonFailure: expected:<Doubling 1 produces [2].> but was:<Doubling 1 produces [3].>
      at org.junit.Assert.assertEquals(Assert.java:117)
      at org.junit.Assert.assertEquals(Assert.java:146)
      at HandlerTests.handleRequest1(HandlerTests.java:13) <-- this the line of failing assertion

FAILURES!!!
Tests run: 2,  Failures: 1
```


lib
hamcrest-core-1.3....
junit-4.13.2.jar
Methods.java
MethodsTests.java

Which tests passed? Which failed?

handleRequest2 passedhandleRequest1 failed because 1 + 2

What other tests would be important to write? Any way to structure the choice of tests?

consider the possibilities of what can be inputted

How could we make writing future tests easier?

Rather than "copy+paste", you could write a method that creates the new handler(args)

change the name to relevant testing case

What would we do to take the file that defines `DoubleHandler` and create a Java program that starts a server using that handler?

Name:	**PID:**

```
// Problem statement: write and test a method median that takes a double[] and produces the
// median number in that array.
```

Input	Expected output
{ 4.0, 5.0, 6.0 }	5.0
{ -1.0, 3.0, 0.0}	0.0
{5.0}	5.0
{2.0, 3.0}	2.5
{}	exception?
{-1.0, -3.0}	-2.0
null	exception?

```
// Consider the following implementations of median:

double medianA(double[] nums) {   Lookup middle
  return nums[nums.length / 2];
}

double medianB(double[] nums) {   Lookup middle but handle even
  int len = nums.length;
  if(len % 2 == 1) { return nums[len / 2]; }
  else { return (nums[len / 2] + nums[len / 2 + 1]) / 2;
}

double medianC(double[] nums) {
  double total = 0;
                                Just mean/average
  for(double d: nums) {
    total += d;
  }
  return total / nums.length;
}

double medianD(double[] nums) {   Sorts but doesn't account for middle
  // make a copy to avoid sorting over the old copy
  double[] cloned = nums.clone();
  Arrays.sort(cloned);
  return nums[cloned.length / 2];
}
```