IT 314

LAB 7

Name :VAIBHAV JINDAL
ID: 202001073

SECTION A

Based on the input ranges, we can identify the following equivalence classes:
Valid dates: The input triple (day, month, year) that represents a valid date in the Gregorian calendar, such as (3, 4, 1995).
Invalid dates: The input triple (day, month, year) that represents an invalid date, such as (31, 2, 2022) or (29, 2, 1900).
Out of range dates: The input triple (day, month, year) that are outside the allowed ranges, such as (0, 5, 2010) or (15, 13, 2005). Based on these equivalence classes, we can design the following test cases:
Tester Action and Input Data Expected Outcome

Valid dates:

Calculate previous date for (15, 10, 2022) 14, 10, 2022
Calculate previous date for (1, 1, 2015) 31, 12, 2014
Calculate previous date for (31, 3, 2000) 30, 3, 2000
Invalid dates:

Calculate previous date for (29, 2, 2022) Invalid date
Calculate previous date for (31, 4, 2010) Invalid date
Calculate previous date for (30, 2, 2000) Invalid date
Out of range dates:

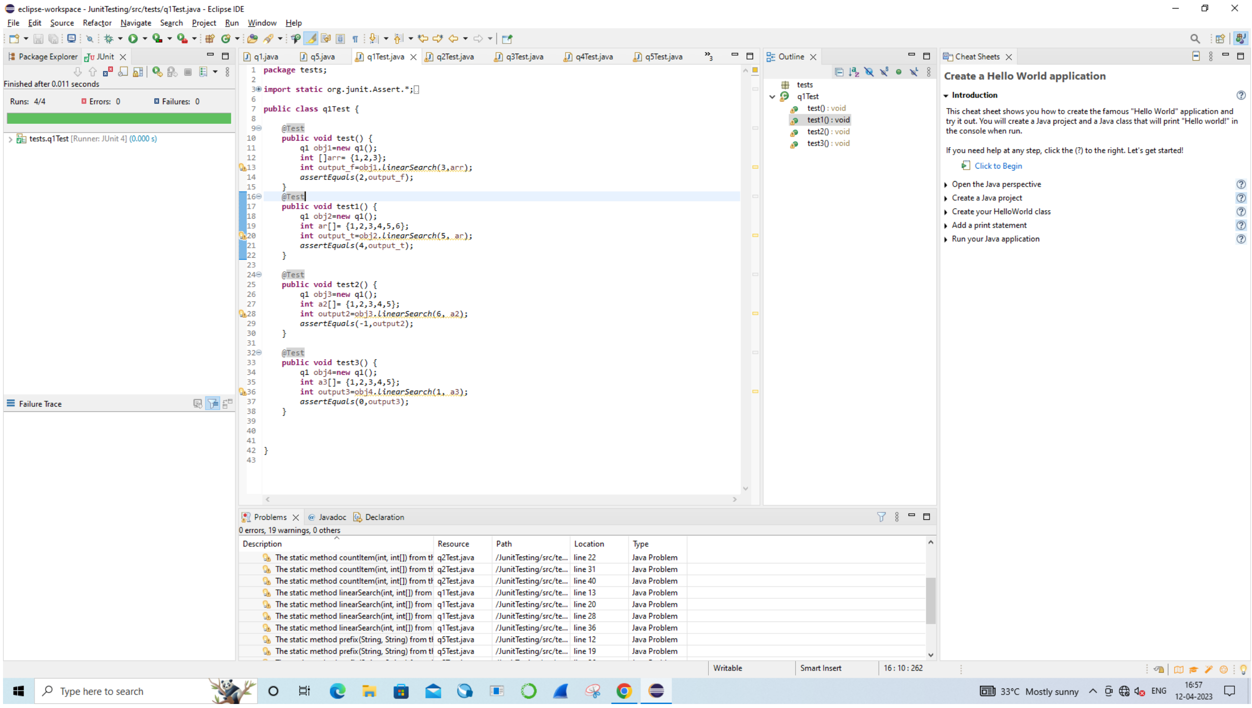
Calculate previous date for (0, 5, 2010) Invalid date
Calculate previous date for (15, 13, 2005) Invalid date
Calculate previous date for (31, 12, 1899) Invalid date
Boundary Value Analysis:
Using boundary value analysis, we can identify the following boundary test cases:

The earliest possible date: (1, 1, 1900)

The latest possible date: (31, 12, 2015)
The earliest day of each month: (1, 1, 2000), (1, 2, 2000), (1, 3, 2000),..., (1, 12, 2000)
The latest day of each month: (31, 1, 2000), (28, 2, 2000), (31, 3, 2000),..., (31, 12, 2000)
Leap year day: (29, 2, 2000)
Invalid leap year day: (29, 2, 1900)
One day before earliest date: (31, 12, 1899)
One day after latest date: (1, 1, 2016)
Based on these boundary test cases, we can design the following test cases:
Tester Action and Input Data Expected Outcome

Boundary Test Cases:

Calculate previous date for (1, 1, 1900) Invalid date
Calculate previous date for (31, 12, 2015) 30, 12, 2015
Calculate previous date for (1, 1, 2000) 31, 12, 1999
Calculate previous date for (31, 1, 2000) 30, 1, 2000
Calculate previous date for (29, 2, 2000) 28, 2, 2000
Calculate previous date for (29, 2, 1900) Invalid date
Calculate previous



Finished after 0.011 seconds

Runs: 4/4 Errors: 0 Failures: 0

tests.q1Test [Runner: JUnit 4] (0.000 s)

```java
package tests;

import static org.junit.Assert.*;

public class q1Test {

    @Test
    public void test() {
        q1 obj1=new q1();
        int []arr= {1,2,3};
        int output_f=obj1.linearSearch(3,arr);
        assertEquals(2,output_f);
    }
    @Test
    public void test1() {
        q1 obj2=new q1();
        int ar[]= {1,2,3,4,5,6};
        int output_t=obj2.linearSearch(5, ar);
        assertEquals(4,output_t);
    }

    @Test
    public void test2() {
        q1 obj3=new q1();
        int a2[]= {1,2,3,4,5};
        int output2=obj3.linearSearch(6, a2);
        assertEquals(-1,output2);
    }

    @Test
    public void test3() {
        q1 obj4=new q1();
        int a3[]= {1,2,3,4,5};
        int output3=obj4.linearSearch(1, a3);
        assertEquals(0,output3);
    }


}
```

tests
q1Test
test() : void
test1() : void
test2() : void
test3() : void

Create a Hello World application

Introduction

This cheat sheet shows you how to create the famous "Hello World" application and try it out. You will create a Java project and a Java class that will print "Hello world!" in the console when run.

If you need help at any step, click the (?) to the right. Let's get started!

Click to Begin

- Open the Java perspective
- Create a Java project
- Create your HelloWorld class
- Add a print statement
- Run your Java application

0 errors, 19 warnings, 0 others

Description	Resource	Path	Location	Type
The static method countItem(int, int[]) from th	q2Test.java	/JunitTesting/src/te...	line 22	Java Problem
The static method countItem(int, int[]) from th	q2Test.java	/JunitTesting/src/te...	line 31	Java Problem
The static method countItem(int, int[]) from th	q2Test.java	/JunitTesting/src/te...	line 40	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 13	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 20	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 28	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 36	Java Problem
The static method prefix(String, String) from th	q5Test.java	/JunitTesting/src/te...	line 12	Java Problem
The static method prefix(String, String) from th	q5Test.java	/JunitTesting/src/te...	line 19	Java Problem

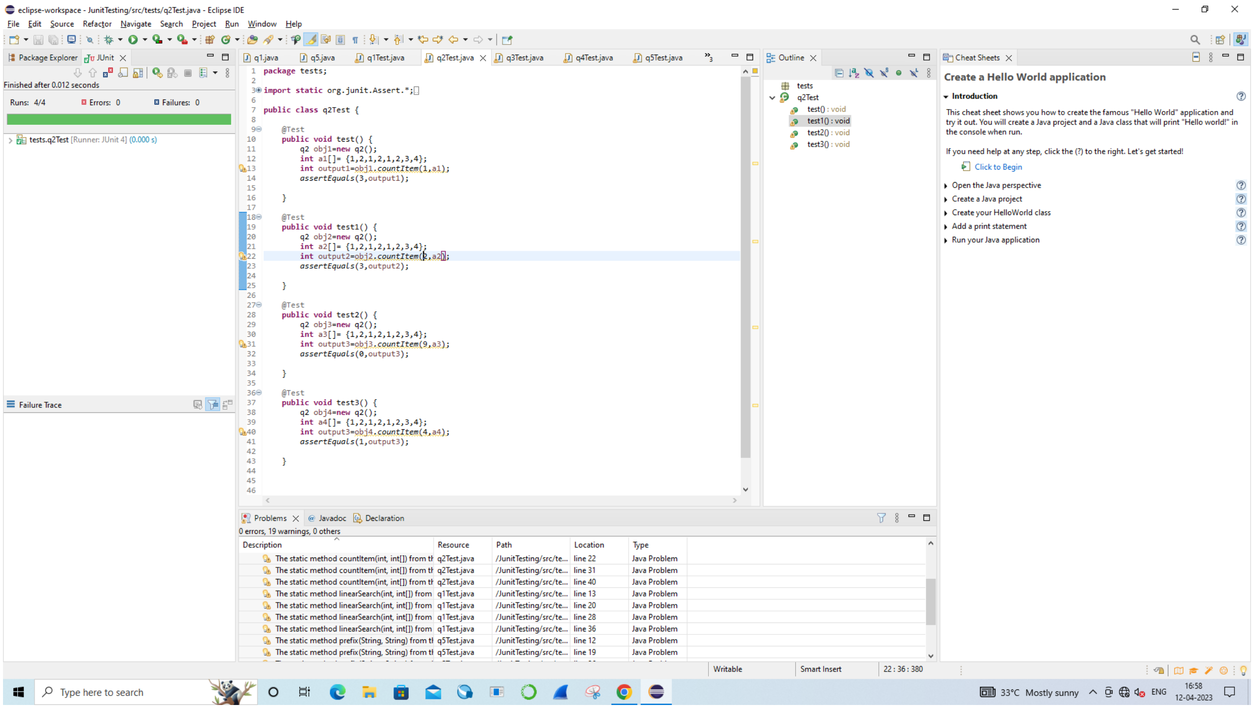
Writable Smart Insert 16 : 10 : 262

33°C Mostly sunny ENG 16:57 12-04-2023

Values	Expected Output	Equivalent Output
5,{1,2,3,4,5,6}	4	4
6,{1,2,3,4,5}	-1	-1

Boundary Case Analysis

Values	Expected Output	Equivalent output
3,{1,2,3}	2	2
1,{1,2,3,4,5}	0	0



Finished after 0.012 seconds

Runs: 4/4   Errors: 0   Failures: 0

tests.q2Test [Runner: JUnit 4] (0.000 s)

```java
package tests;

import static org.junit.Assert.*;

public class q2Test {

    @Test
    public void test() {
        q2 obj1=new q2();
        int a1[]= {1,2,1,2,1,2,3,4};
        int output1=obj1.countItem(1,a1);
        assertEquals(3,output1);

    }

    @Test
    public void test1() {
        q2 obj2=new q2();
        int a2[]= {1,2,1,2,1,2,3,4};
        int output2=obj2.countItem(2,a2);
        assertEquals(3,output2);

    }

    @Test
    public void test2() {
        q2 obj3=new q2();
        int a3[]= {1,2,1,2,1,2,3,4};
        int output3=obj3.countItem(9,a3);
        assertEquals(0,output3);

    }

    @Test
    public void test3() {
        q2 obj4=new q2();
        int a4[]= {1,2,1,2,1,2,3,4};
        int output3=obj4.countItem(4,a4);
        assertEquals(1,output3);

    }
```

tests
q2Test
test() : void
test1() : void
test2() : void
test3() : void

Create a Hello World application

Introduction

This cheat sheet shows you how to create the famous "Hello World" application and try it out. You will create a Java project and a Java class that will print "Hello world!" in the console when run.

If you need help at any step, click the (?) to the right. Let's get started!

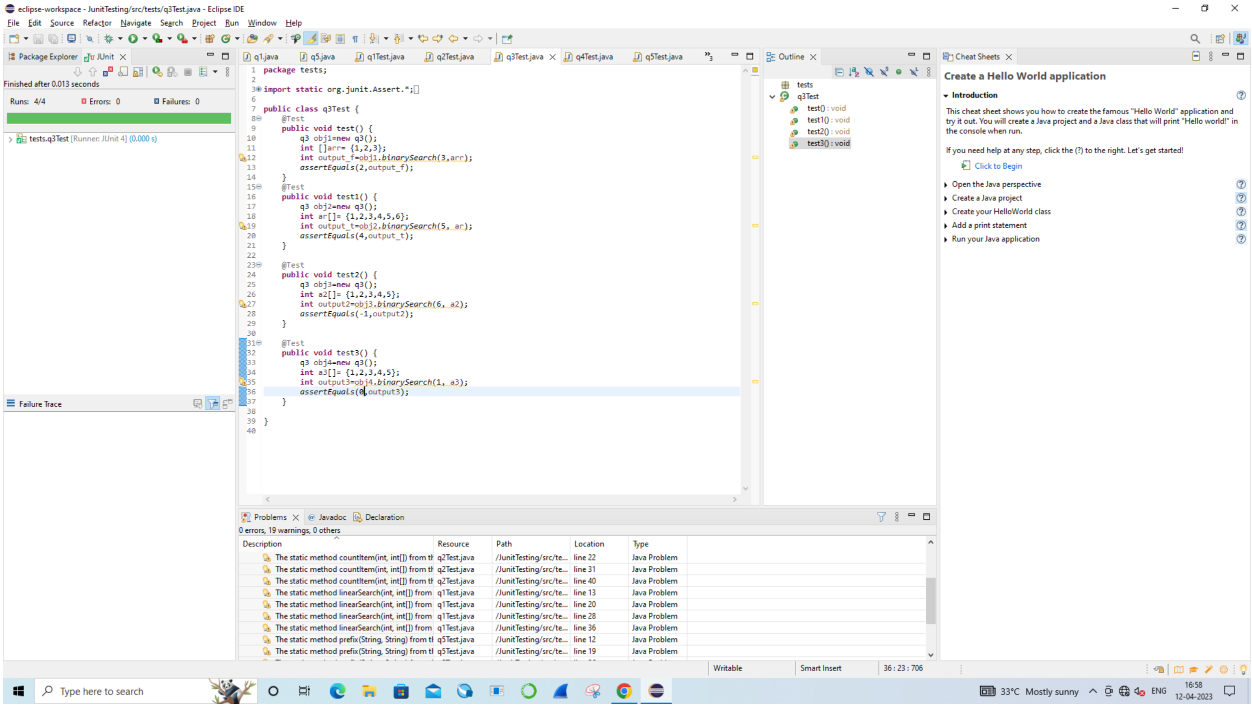
Click to Begin

- Open the Java perspective
- Create a Java project
- Create your HelloWorld class
- Add a print statement
- Run your Java application

0 errors, 19 warnings, 0 others

Description	Resource	Path	Location	Type
The static method countItem(int, int[]) from th	q2Test.java	/JunitTesting/src/te...	line 22	Java Problem
The static method countItem(int, int[]) from th	q2Test.java	/JunitTesting/src/te...	line 31	Java Problem
The static method countItem(int, int[]) from th	q2Test.java	/JunitTesting/src/te...	line 40	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 13	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 20	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 28	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 36	Java Problem
The static method prefix(String, String) from th	q5Test.java	/JunitTesting/src/te...	line 12	Java Problem
The static method prefix(String, String) from th	q5Test.java	/JunitTesting/src/te...	line 19	Java Problem

Values	Expected Output	Equivalent output
1,{1,2,1,2,1,2,3,4}	3	3
2,{1,2,1,2,1,2,3,4}	3	3
9,{1,2,1,2,1,2,3,4}	0	0
4,{1,2,1,2,1,2,3,4}	1	1



Finished after 0.013 seconds

Runs: 4/4 Errors: 0 Failures: 0

tests.q3Test [Runner: JUnit 4] (0.000 s)

```java
package tests;

import static org.junit.Assert.*;

public class q3Test {
    @Test
    public void test() {
        q3 obj1=new q3();
        int []arr= {1,2,3};
        int output_f=obj1.binarySearch(3,arr);
        assertEquals(2,output_f);
    }
    @Test
    public void test1() {
        q3 obj2=new q3();
        int ar[]= {1,2,3,4,5,6};
        int output_t=obj2.binarySearch(5, ar);
        assertEquals(4,output_t);
    }

    @Test
    public void test2() {
        q3 obj3=new q3();
        int a2[]= {1,2,3,4,5};
        int output2=obj3.binarySearch(6, a2);
        assertEquals(-1,output2);
    }

    @Test
    public void test3() {
        q3 obj4=new q3();
        int a3[]= {1,2,3,4,5};
        int output3=obj4.binarySearch(1, a3);
        assertEquals(0,output3);
    }

}
```

tests
q3Test
test() : void
test1() : void
test2() : void
test3() : void

Create a Hello World application

Introduction

This cheat sheet shows you how to create the famous "Hello World" application and try it out. You will create a Java project and a Java class that will print "Hello world!" in the console when run.

If you need help at any step, click the (?) to the right. Let's get started!

Click to Begin

- Open the Java perspective
- Create a Java project
- Create your HelloWorld class
- Add a print statement
- Run your Java application

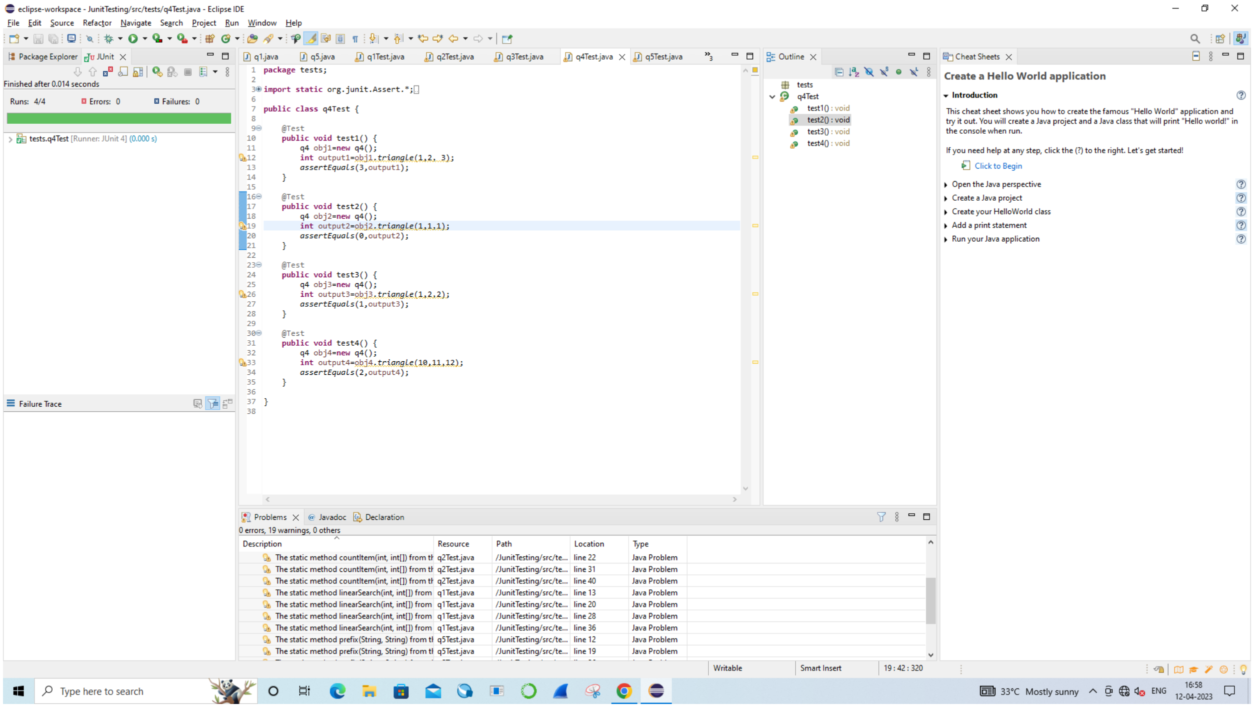
0 errors, 19 warnings, 0 others

Description	Resource	Path	Location	Type
The static method countItem(int, int[]) from th	q2Test.java	/JunitTesting/src/te...	line 22	Java Problem
The static method countItem(int, int[]) from th	q2Test.java	/JunitTesting/src/te...	line 31	Java Problem
The static method countItem(int, int[]) from th	q2Test.java	/JunitTesting/src/te...	line 40	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 13	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 20	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 28	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 36	Java Problem
The static method prefix(String, String) from th	q5Test.java	/JunitTesting/src/te...	line 12	Java Problem
The static method prefix(String, String) from th	q5Test.java	/JunitTesting/src/te...	line 19	Java Problem

Values	Expected Output	Equivalent Output
5,{1,2,3,4,5,6}	4	4
6,{1,2,3,4,5}	-1	-1

Boundary Case Analysis

Values	Expected Output	Equivalent output
3,{1,2,3}	2	2
1,{1,2,3,4,5}	0	0



Finished after 0.014 seconds

Runs: 4/4 Errors: 0 Failures: 0

tests.q4Test [Runner: JUnit 4] (0.000 s)

```java
package tests;

import static org.junit.Assert.*;

public class q4Test {

    @Test
    public void test1() {
        q4 obj1=new q4();
        int output1=obj1.triangle(1,2, 3);
        assertEquals(3,output1);
    }

    @Test
    public void test2() {
        q4 obj2=new q4();
        int output2=obj2.triangle(1,1,1);
        assertEquals(0,output2);
    }

    @Test
    public void test3() {
        q4 obj3=new q4();
        int output3=obj3.triangle(1,2,2);
        assertEquals(1,output3);
    }

    @Test
    public void test4() {
        q4 obj4=new q4();
        int output4=obj4.triangle(10,11,12);
        assertEquals(2,output4);
    }

}
```

0 errors, 19 warnings, 0 others

Description	Resource	Path	Location	Type
The static method countItem(int, int[]) from th	q2Test.java	/JunitTesting/src/te...	line 22	Java Problem
The static method countItem(int, int[]) from th	q2Test.java	/JunitTesting/src/te...	line 31	Java Problem
The static method countItem(int, int[]) from th	q2Test.java	/JunitTesting/src/te...	line 40	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 13	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 20	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 28	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 36	Java Problem
The static method prefix(String, String) from th	q5Test.java	/JunitTesting/src/te...	line 12	Java Problem
The static method prefix(String, String) from th	q5Test.java	/JunitTesting/src/te...	line 19	Java Problem

Create a Hello World application

Introduction

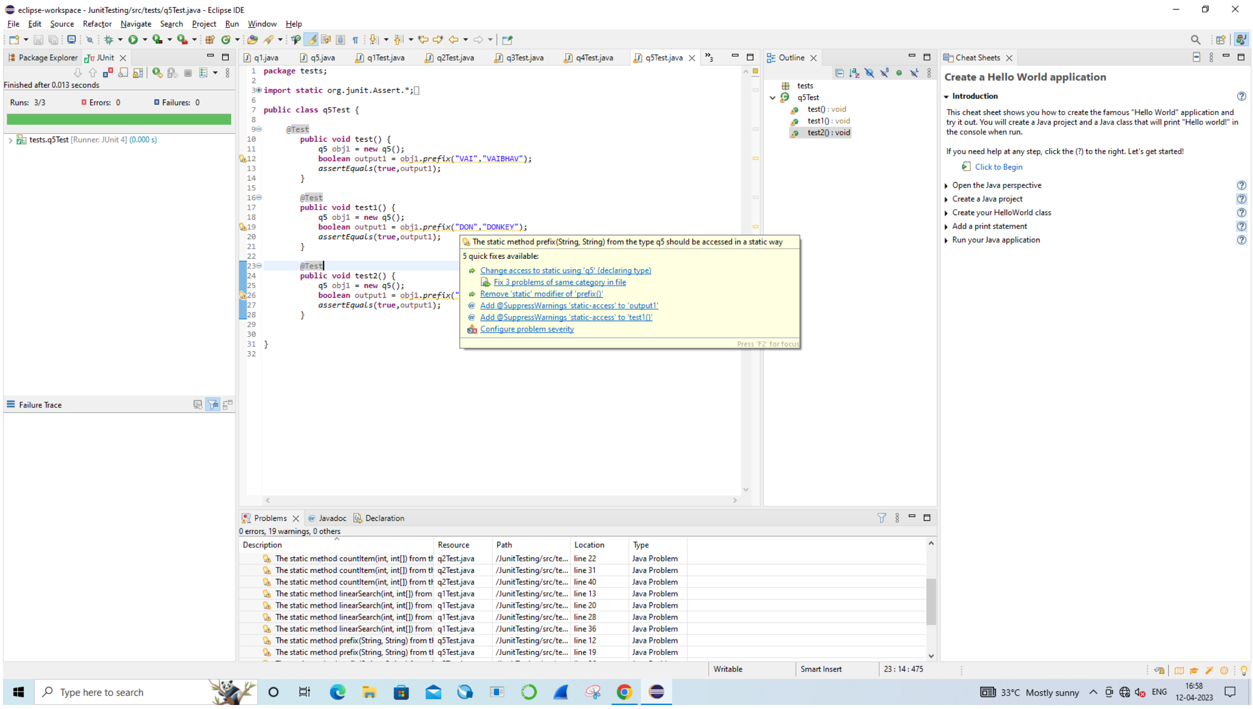
This cheat sheet shows you how to create the famous "Hello World" application and try it out. You will create a Java project and a Java class that will print "Hello world!" in the console when run.

If you need help at any step, click the (?) to the right. Let's get started!

Click to Begin

- Open the Java perspective
- Create a Java project
- Create your HelloWorld class
- Add a print statement
- Run your Java application

Values	Expected Output	Equivalent Output
1,2,3	3	3
1,1,1	0	0
1,2,2	1	1
10,11,12	2	2



eclipse-workspace - JunitTesting/src/tests/q5Test.java - Eclipse IDE

Finished after 0.013 seconds

Runs: 3/3 Errors: 0 Failures: 0

tests.q5Test [Runner: JUnit 4] (0.000 s)

```
package tests;

import static org.junit.Assert.*;

public class q5Test {

	@Test
	public void test() {
		q5 obj1 = new q5();
		boolean output1 = obj1.prefix("VAI","VAIBHAV");
		assertEquals(true,output1);
	}

	@Test
	public void test1() {
		q5 obj1 = new q5();
		boolean output1 = obj1.prefix("DON","DONKEY");
		assertEquals(true,output1);
	}

	@Test
	public void test2() {
		q5 obj1 = new q5();
		boolean output1 = obj1.prefix("
		assertEquals(true,output1);
	}


}
```

The static method prefix(String, String) from the type q5 should be accessed in a static way

5 quick fixes available:

- Change access to static using 'q5' (declaring type)
  - Fix 3 problems of same category in file
- Remove 'static' modifier of 'prefix()'
- Add @SuppressWarnings 'static-access' to 'output1'
- Add @SuppressWarnings 'static-access' to 'test1()'
- Configure problem severity

Press 'F2' for focus

Create a Hello World application

Introduction

This cheat sheet shows you how to create the famous "Hello World" application and try it out. You will create a Java project and a Java class that will print "Hello world!" in the console when run.

If you need help at any step, click the (?) to the right. Let's get started!

Click to Begin

- Open the Java perspective
- Create a Java project
- Create your HelloWorld class
- Add a print statement
- Run your Java application

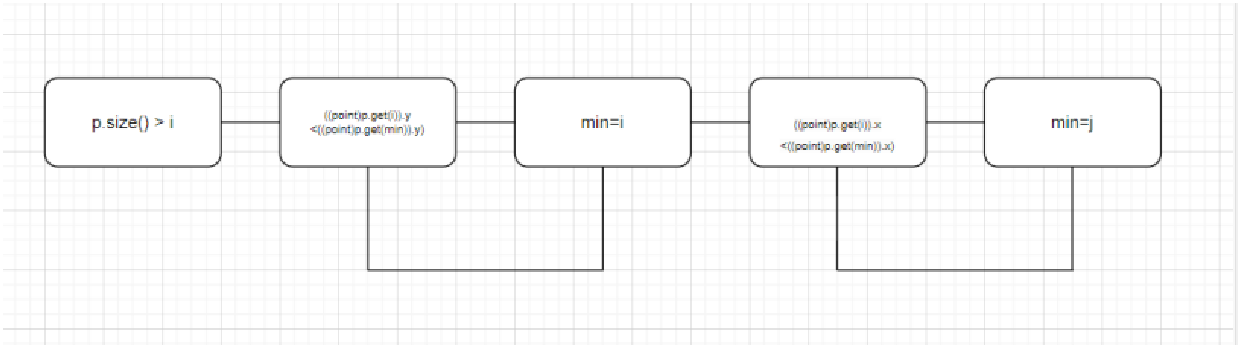
0 errors, 19 warnings, 0 others

Description	Resource	Path	Location	Type
The static method countItem(int, int[]) from th	q2Test.java	/JunitTesting/src/te...	line 22	Java Problem
The static method countItem(int, int[]) from th	q2Test.java	/JunitTesting/src/te...	line 31	Java Problem
The static method countItem(int, int[]) from th	q2Test.java	/JunitTesting/src/te...	line 40	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 13	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 20	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 28	Java Problem
The static method linearSearch(int, int[]) from	q1Test.java	/JunitTesting/src/te...	line 36	Java Problem
The static method prefix(String, String) from th	q5Test.java	/JunitTesting/src/te...	line 12	Java Problem
The static method prefix(String, String) from th	q5Test.java	/JunitTesting/src/te...	line 19	Java Problem

Writable Smart Insert 23 : 14 : 475

Values	Expected Output	Equivalent Output
VAI,VAIBHAV	true	true
DON,DONKEY	true	true
DIDAS,DIO	false	false

SECTION B


p.size() > i
((point)p.get(i)).y <((point)p.get(min)).y)
min=i
((point)p.get(i)).x <((point)p.get(min)).x)
min=j

2. a.Statement Coverage test set:
Test Case 1: p.size() > point i.e. 2 is false
Test Case 2: p.size() > 2 is true
b. Branch Coverage Test Set:
Test Case 1: p.size() > point i.e. 2 is false
Test Case 2: p.size() > 2 is true and loop is executed
Test Case 3: p.size() > 2 is true and loop is not executed
c. Basic Condition Coverage Test Set:
Test Case 1: p.size() > point i.e. 2 is false
Test Case 2: p.size() > 2 is true and loop is executed
Test Case 3: p.size() > 2 is true and loop is not executed
Test Case 4: p.size() > 2 is true and loop is executed twice