



Universidad de las Fuerzas Armadas ESPE

Object Oriented Programming

Teacher: Edison Lascano

Student: Revilla Antonio

Nrc: 14575

Homework #18

Topic: Unit Test to Taxes Lib

```
Test Packages
  ec.edu.espe.unittest.controller
    TaxivaTest.java
  Dependencies
  Test Dependencies
  Java Dependencies
  Project Files
  HWMongoDB
  JavaMongoDB
  Persistence
  SystemZoo
  TaxLib

6 | */
7 | public class TaxIva {
8 |
9 |     public static double calcularIva(double valorProducto, double tasaIva) {
10 |         float result;
11 |         valorProducto = 100;
12 |         tasaIva = 1.12;
13 |         result = (float) (valorProducto * tasaIva);
14 |         System.out.println("IVA --->" + result);
15 |         return result;
16 |     }
17 | }
18 |

Output - Run (HW18)
cd C:\Users\revil\Documents\NetBeansProjects\Unit2\WS34Persistence\HW18; "JAVA_HOME=C:\\Program Files (x86)\\Java 21" cmd /c "%C:\\Program Files\\NetBeans-19\\netbea
Scanning for projects...

-----< com.mycompany:HW18 >-----
Building HW18 1.0-SNAPSHOT
from pom.xml
-----[ jar ]-----

--- resources:3.3.1:resources (default-resources) @ HW18 ---
skip non existing resourceDirectory C:\Users\revil\Documents\NetBeansProjects\Unit2\WS34Persistence\HW18\src\main\resources

--- compiler:3.11.0:compile (default-compile) @ HW18 ---
Nothing to compile - all classes are up to date

--- exec:3.1.0:exec (default-cli) @ HW18 ---
IVA --->112.0
```

```
41 | */
42 | @org.junit.jupiter.api.Test
43 | public void testCalcularIva() {
44 |     System.out.println("CalcularIva");
45 |     double valorProducto = 100.0;
46 |     double tasaIva = 1.12;
47 |     double expectedResult = 112.0;
48 |     double result = TaxIva.calcularIva(valorProducto, tasaIva);
49 |     assertEquals("expected: expectedResult, actual: result, delta: 0.000");
50 | }
51 |
52 | }
53 |

Output Test Results X
com.mycompany:HW18:jar:1.0-SNAPSHOT (Unit) X

Tests passed: 100.00 %
The test passed. (0,045 s)

calculaIva
IVA --->112.0
```

```

39  /**
40   * Test of computeIncomeTax method, of class incomeTax.
41   */
42   @Test
43   public void testComputeIncomeTax() {
44       System.out.println("computeIncomeTax");
45       float monthlyIncome = -1.0F;
46       float expectedResult = 0.0F;
47       float result = incomeTax.computeIncomeTax(monthlyIncome);
48       assertEquals(expected: expectedResult, actual: result, delta: 0.01);
49   }
50
51   @Test
52   public void testComputeIncomeTax_1() {
53       System.out.println("computeIncomeTax");
54       float monthlyIncome = 250.0F;

```

```

19  System.out.pr
20  } else if (netSal
21      incomeTax = r
22  } else if (netSal
23      incomeTax = .
24  } else if (netSal
25      incomeTax = .
26  } else if (netSal
27      incomeTax = .
28  } else if (netSal
29      incomeTax = .
30  } else if (netSal
31      incomeTax = .
32  } else if (netSal
33      incomeTax = .
34  } else if (netSal
35      incomeTax = .

```

Test Results

com.mycompany.HW18.jar:1.0-SNAPSHOT (Unit)

Tests passed: 92.31%

12 tests passed, 1 test failed. (0.057 s)

ec.edu.espe.unitest.controller.incomeTaxTest Failed

testComputeIncomeTax_1 Failed: expected: <0.0> but was: <2656.5>

```

computeIncomeTax
it is not possible to calculate taxes
Income Tax -->0.0

computeIncomeTax
Income Tax -->3.3061427E8

computeIncomeTax
it is not possible to calculate taxes
Income Tax -->0.0

computeIncomeTax
Income Tax -->2656.5

computeIncomeTax
Income Tax -->10094.7

computeIncomeTax
Income Tax -->11688.6

computeIncomeTax
Income Tax -->1511.7859

computeIncomeTax
Income Tax -->1313.9404

```

```

42   @Test
43   public void testComputeIncomeTax() {
44       System.out.println("computeIncomeTax");
45       float monthlyIncome = -1.0F;
46       float expectedResult = 0.0F;
47       float result = incomeTax.computeIncomeTax(monthlyIncome);
48       assertEquals(expected: expectedResult, actual: result, delta: 0);
49   }
50
51   @Test
52   public void testComputeIncomeTax_1() {
53       System.out.println("computeIncomeTax");
54       float monthlyIncome = 250.0F;
55       float expectedResult = 0.0F;
56       float result = incomeTax.computeIncomeTax(monthlyIncome);
57       assertEquals(expected: expectedResult, actual: result, delta: 0);
58   }
59
60   @Test
61   public void testComputeIncomeTax_2() {
62       System.out.println("computeIncomeTax");
63       float monthlyIncome = 950.0F;
64       float expectedResult = 10094.7F;
65       float result = incomeTax.computeIncomeTax(monthlyIncome);
66       assertEquals(expected: expectedResult, actual: result, delta: 0);
67   }
68
69   @Test
70   public void testComputeIncomeTax_3() {
71       System.out.println("computeIncomeTax");
72       float monthlyIncome = 1100.0F;
73       float expectedResult = 11688.6F;
74       float result = incomeTax.computeIncomeTax(monthlyIncome);
75       assertEquals(expected: expectedResult, actual: result, delta: 0);
76   }
77
78   @Test
79   public void testComputeIncomeTax_4() {
80       System.out.println("computeIncomeTax");
81       float monthlyIncome = 2510.50F;
82       float expectedResult = 1511.7859F;
83       float result = incomeTax.computeIncomeTax(monthlyIncome);
84       assertEquals(expected: expectedResult, actual: result, delta: 0);
85   }
86
87   @Test

```

```

13  float annualIncome = monthlyIncome * 12;
14
15  float netSalary = annualIncome - (annualIncome * 0.1145f);
16
17  if (netSalary < 0) {
18      incomeTax = 0;
19      System.out.println("it is not possible to calculate taxes");
20  } else if (netSalary <= 11722) {
21      incomeTax = netSalary;
22  } else if (netSalary <= 14930) {
23      incomeTax = (netSalary - 11722) * 0.5f + 0;
24  } else if (netSalary <= 19385) {
25      incomeTax = (netSalary - 14930) * 0.10f + 160;
26  } else if (netSalary <= 25638) {
27      incomeTax = (netSalary - 19385) * 0.12f + 606;
28  } else if (netSalary <= 33738) {
29      incomeTax = (netSalary - 25638) * 0.15f + 1356;
30  } else if (netSalary <= 44721) {
31      incomeTax = (netSalary - 33738) * 0.2f + 2571;
32  } else if (netSalary <= 59537) {
33      incomeTax = (netSalary - 44721) * 0.25f + 4768;
34  } else if (netSalary <= 79388) {
35      incomeTax = (netSalary - 59537) * 0.3f + 8472;
36  } else if (netSalary <= 105580) {
37      incomeTax = (netSalary - 79388) * 0.35f + 14427;
38  } else {
39      incomeTax = (netSalary - 105580) * 0.37f + 23594;
40  }
41
42  System.out.println("Income Tax -->" + incomeTax);
43  return incomeTax;
44  }
45
46  }

```

com.mycompany.HW18.jar:1.0-SNAPSHOT (Unit) x

Tests passed: 76.92 %

10 tests passed, 3 tests failed. (0.059 s)

- ec.edu.espe.unittest.controller.incomeTaxTest Failed
 - testComputeIncomeTax_1 Failed: expected: <0.0> but was: <2656.5>
 - testComputeIncomeTax_5 Failed: expected: <1313.9408> but was: <1313.9404>
 - testComputeIncomeTax_6 Failed: expected: <6739.295> but was: <6739.285>

```
computeIncomeTax
it is not possible to calculate taxes
Income Tax -->0.0

computeIncomeTax
Income Tax -->3.3061427E8

computeIncomeTax
it is not possible to calculate taxes
Income Tax -->0.0

computeIncomeTax
Income Tax -->2656.5

computeIncomeTax
Income Tax -->10094.7

computeIncomeTax
Income Tax -->11688.6

computeIncomeTax
Income Tax -->1511.7859

computeIncomeTax
Income Tax -->1313.9404

computeIncomeTax
Income Tax -->6739.285

computeIncomeTax
Income Tax -->14265.174

computeIncomeTax
Income Tax -->28763.074

computeIncomeTax
Income Tax -->2732.0

calcularIva
IVA --->112.0
```

```
48 assertEquals(expected:expResult, actual: result, delta: 0.01);
49
50
51 @Test
52 public void testComputeIncomeTax_1() {
53     System.out.println("computeIncomeTax");
54     float monthlyIncome = 250.0F;
55     float expResult = 2656.5F;
56     float result = incomeTax.computeIncomeTax(monthlyIncome);
57     assertEquals(expected:expResult, actual: result, delta: 0.01);
58 }
59
60 @Test
61 public void testComputeIncomeTax_2() {
62     System.out.println("computeIncomeTax");
63     float monthlyIncome = 950.0F;
64     float expResult = 10094.7F;
65     float result = incomeTax.computeIncomeTax(monthlyIncome);
66     assertEquals(expected:expResult, actual: result, delta: 0.01);
67 }
68
69 @Test
70 public void testComputeIncomeTax_3() {
71
72     float annualIncome = monthlyIncome * 12;
73
74     float netSalary = annualIncome - (annualIncome * 0.1145f);
75
76     if (netSalary < 0) {
77         incomeTax = 0;
78         System.out.println("it is not possible to calculate taxes");
79     } else if (netSalary <= 11722) {
80         incomeTax = netSalary;
81     } else if (netSalary <= 14930) {
82         incomeTax = ((netSalary - 11722) * 0.5f) + 0;
83     } else if (netSalary <= 19385) {
84         incomeTax = ((netSalary - 14930) * 0.10f) + 160;
85     } else if (netSalary <= 25638) {
86         incomeTax = ((netSalary - 19385) * 0.12f) + 606;
87     } else if (netSalary <= 33738) {
88         incomeTax = ((netSalary - 25638) * 0.15f) + 1356;
89     } else if (netSalary <= 44721) {
90         incomeTax = ((netSalary - 33738) * 0.2f) + 2571;
91     } else if (netSalary <= 59537) {
92         incomeTax = ((netSalary - 44721) * 0.25f) + 4768;
93     } else if (netSalary <= 79388) {
94         incomeTax = ((netSalary - 59537) * 0.3f) + 8472;
95     } else if (netSalary <= 105580) {
96
97     }
98 }
```

Test Results X

com.mycompany.HW18.jar:1.0-SNAPSHOT (Unit) x

Tests passed: 100.00 %

All 13 tests passed (0.057 s)

```
computeIncomeTax
it is not possible to calculate taxes
Income Tax -->0.0

computeIncomeTax
Income Tax -->3.3061427E8

computeIncomeTax
it is not possible to calculate taxes
Income Tax -->0.0

computeIncomeTax
Income Tax -->2656.5

computeIncomeTax
Income Tax -->10094.7

computeIncomeTax
Income Tax -->11688.6

computeIncomeTax
Income Tax -->1511.7859
```