

SSN:

The screenshot shows two separate runs of the CSDC437Lab5RegexTests project in IntelliJ IDEA. Both runs show 9 tests passed in 21ms, with a process finished exit code of 0.

Run 1 (Top):

```
19     @Test
20     public void SSN_valid_format_prints_valid_response() {
21         String validSSN = "123456789";
22         CSCD437RegExMethods.SSNPattern(validSSN, new PrintStream(testOut));
23         assertFalse(testOut.toString().contains("NOT"));
24     }
25
26     @Test
27     public void SSN_valid_format_prints_valid_response2() {
28         String validSSN = "123-45-6789";
29         CSCD437RegExMethods.SSNPattern(validSSN, new PrintStream(testOut));
30         assertFalse(testOut.toString().contains("NOT"));
31     }
32
33     @Test
34     public void SSN_valid_format_prints_valid_response3() {
35         String validSSN = "123 45 6789";
36         CSCD437RegExMethods.SSNPattern(validSSN, new PrintStream(testOut));
37         assertFalse(testOut.toString().contains("NOT"));
38     }
39
```

Run 2 (Bottom):

```
40     @Test
41     public void SSN_valid_format_prints_valid_response4() {
42         String validSSN = "123-45 6789";
43         CSCD437RegExMethods.SSNPattern(validSSN, new PrintStream(testOut));
44         assertFalse(testOut.toString().contains("NOT"));
45     }
46
47     @Test
48     public void SSN_valid_format_prints_valid_response5() {
49         String validSSN = "123 45-6789";
50         CSCD437RegExMethods.SSNPattern(validSSN, new PrintStream(testOut));
51         assertFalse(testOut.toString().contains("NOT"));
52     }
53
54     @Test
55     public void SSN_valid_format_prints_valid_response6() {
56         String validSSN = "12345-6789";
57         CSCD437RegExMethods.SSNPattern(validSSN, new PrintStream(testOut));
58         assertFalse(testOut.toString().contains("NOT"));
59     }
60
```

Both runs show the same test results and execution time of 21ms.

```

    public void SSNPattern_invalid_format_prints_invalid_response() {
        String validSSN = "12-345-6789";
        CSCD437RegExMethods.SSNPattern(validSSN, new PrintStream(testOut));
        assertTrue(testOut.toString().contains("NOT"));
    }

    public void SSNPattern_invalid_format_prints_invalid_response2() {
        String validSSN = "123--45-6789";
        CSCD437RegExMethods.SSNPattern(validSSN, new PrintStream(testOut));
        assertTrue(testOut.toString().contains("NOT"));
    }

    public void SSNPattern_invalid_format_prints_invalid_response3() {
        String validSSN = "123.45.6789";
        CSCD437RegExMethods.SSNPattern(validSSN, new PrintStream(testOut));
        assertTrue(testOut.toString().contains("NOT"));
    }
}

```

Run: CSCD437Lab5RegexTests > Tests passed: 9 of 9 tests – 21ms

Process finished with exit code 0

Phone Number:

```

    public void PhoneNumber_valid_format_prints_valid_response() {
        String validPhoneNumber = "+509 123-4567";
        CSCD437RegExMethods.phonePattern(validPhoneNumber, new PrintStream(testOut));
        assertFalse(testOut.toString().contains("NOT"));
    }

    public void PhoneNumber_valid_format_prints_valid_response2() {
        String validPhoneNumber = "509-123-4567";
        CSCD437RegExMethods.phonePattern(validPhoneNumber, new PrintStream(testOut));
        assertFalse(testOut.toString().contains("NOT"));
    }

    public void PhoneNumber_valid_format_prints_valid_response3() {
        String validPhoneNumber = "123 456 7890";
        CSCD437RegExMethods.phonePattern(validPhoneNumber, new PrintStream(testOut));
        assertFalse(testOut.toString().contains("NOT"));
    }

    public void PhoneNumber_valid_format_prints_valid_response4() {
        String validPhoneNumber = "+1 509-123-4567";
        CSCD437RegExMethods.phonePattern(validPhoneNumber, new PrintStream(testOut));
        assertFalse(testOut.toString().contains("NOT"));
    }
}

```

Run: CSCD437Lab5RegexTests > Tests passed: 8 of 8 tests – 14ms

Process finished with exit code 0

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project:** TL-Lab5
- Open File:** CSDC437Lab5RegexTests.java
- Code Snippet:** The code contains several test methods for validating phone numbers and SSN patterns. The methods include:
 - `public void PhoneNumber_valid_format_prints_valid_response4()`
 - `public void PhoneNumber_valid_format_prints_valid_response5()`
 - `public void SSNPattern_invalid_format_prints_invalid_response()`
 - `public void SSNPattern_invalid_format_prints_invalid_response2()`
- Run Tab:** Shows the test results:
 - Tests passed: 8 of 8 tests - 14 ms
 - Test cases listed: SSNPattern_invalid_format_10ms, SSNPattern_invalid_format_pr_1ms, SSNPattern_invalid_format_pr_1ms, PhoneNumber_valid_format_1ms, PhoneNumber_valid_format_1ms, PhoneNumber_valid_format_prints, PhoneNumber_valid_format_prints, PhoneNumber_valid_format_prints.
- Bottom Status Bar:** Tests passed: 8 (2 minutes ago)

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project:** TL-Lab5
- File:** CSCD437Lab5RegexTests.java
- Code Snippet:**

```
54
55     @Test
56     public void SSNPattern_invalid_format_prints_invalid_response() {
57         String validPhoneNumber = "1 (509) 123-4567";
58         CSCD437RegExMethods.phonePattern(validPhoneNumber, new PrintStream(testOut));
59         assertTrue(testOut.toString().contains("NOT"));
60     }
61
62     @Test
63     public void SSNPattern_invalid_format_prints_invalid_response2() {
64         String validPhoneNumber = "509.123.4567";
65         CSCD437RegExMethods.phonePattern(validPhoneNumber, new PrintStream(testOut));
66         assertTrue(testOut.toString().contains("NOT"));
67     }
68
69     @Test
70     public void SSNPattern_invalid_format_prints_invalid_response3() {
71         String validPhoneNumber = "123--456-7890";
72         CSCD437RegExMethods.phonePattern(validPhoneNumber, new PrintStream(testOut));
73         assertTrue(testOut.toString().contains("NOT"));
74     }
75 }
```

- Run:** CSCD437Lab5RegexTests
- Output:** Tests passed: 8 of 8 tests - 14 ms
- Structure:** Shows the class hierarchy and methods.
- Bottom Bar:** Version Control, Run, TODO, Problems, Terminal, Profiler, Services, Build.

Email Address:

The screenshots show the IntelliJ IDEA interface with two Java files open:

- CSCD437Lab5RegexTests.java**: Contains three test methods:
 - `@Test public void emailPattern_valid_format_prints_valid_response() { String validEmail = "t.holmquist@gmail.com"; CSCD437RegExMethods.emailPattern(validEmail, new PrintStream(testOut)); assertFalse(testOut.toString().contains("NOT")); }`
 - `@Test public void emailPattern_valid_format_prints_valid_response2() { String validEmail = "t.holmquist@gmail.com"; CSCD437RegExMethods.emailPattern(validEmail, new PrintStream(testOut)); assertFalse(testOut.toString().contains("NOT")); }`
 - `@Test public void emailPattern_valid_format_prints_valid_response3() { String validEmail = "t.holmquist.9080@ghost.com"; CSCD437RegExMethods.emailPattern(validEmail, new PrintStream(testOut)); assertFalse(testOut.toString().contains("NOT")); }`
- CSCD437Lab5RegexTests.java**: Contains five test methods:
 - `@Test public void emailPattern_valid_format_prints_valid_response4() { String validEmail = "tholmquist9080@gmail.com"; CSCD437RegExMethods.emailPattern(validEmail, new PrintStream(testOut)); assertFalse(testOut.toString().contains("NOT")); }`
 - `@Test public void emailPattern_valid_format_prints_valid_response5() { String validEmail = "t.holmquist@yahoo.com"; CSCD437RegExMethods.emailPattern(validEmail, new PrintStream(testOut)); assertFalse(testOut.toString().contains("NOT")); }`

Both editors show a green checkmark icon next to the test names in the run output, indicating that all tests have passed.

The screenshot shows the IntelliJ IDEA interface with the project 'TL-Lab5' open. The 'Run' tool window at the bottom displays the results of a test run for 'CSCD437Lab5RegexTests'. The results show 9 tests passed in 15ms. The test cases listed are: emailPattern_valid_format_pr_3ms, emailPattern_valid_format_prints_v, emailPattern_valid_format_prints_v, emailPattern_valid_format_pr_1ms, emailPattern_invalid_format_p_1ms, emailPattern_valid_format_prints_v, emailPattern_invalid_format_prints_, emailPattern_invalid_format_prints_, and emailPattern_invalid_format_prints_. The status bar at the bottom right indicates the time as 82:51, LF, UTF-8, 4 spaces.

```
59     @Test
60     public void emailPattern_invalid_format_prints_invalid_response() {
61         String validEmail = "tholmquist789wQ";
62         CSCD437RegExMethods.emailPattern(validEmail, new PrintStream(testOut));
63         assertTrue(testOut.toString().contains("NOT"));
64     }
65
66     @Test
67     public void emailPattern_invalid_format_prints_invalid_response2() {
68         String validEmail = "t.jolmquist@skidai";
69         CSCD437RegExMethods.emailPattern(validEmail, new PrintStream(testOut));
70         assertTrue(testOut.toString().contains("NOT"));
71     }
72
73     @Test
74     public void emailPattern_invalid_format_prints_invalid_response3() {
75         String validEmail = "tholmquist_gj@yahoo.com";
76         CSCD437RegExMethods.emailPattern(validEmail, new PrintStream(testOut));
77         assertTrue(testOut.toString().contains("NOT"));
78     }
79 }
```

The screenshot shows the IntelliJ IDEA interface with the project 'TL-Lab5' open. The 'Run' tool window at the bottom displays the results of a test run for 'CSCD437Lab5RegexTests'. The results show 9 tests passed in 15ms. The test cases listed are: emailPattern_valid_format_pr_3ms, emailPattern_valid_format_prints_v, emailPattern_valid_format_prints_v, emailPattern_valid_format_pr_1ms, emailPattern_invalid_format_p_1ms, emailPattern_valid_format_prints_v, emailPattern_invalid_format_prints_, emailPattern_invalid_format_prints_, and emailPattern_invalid_format_prints_. The status bar at the bottom right indicates the time as 82:51, LF, UTF-8, 4 spaces.

```
74     public void emailPattern_invalid_format_prints_invalid_response3() {
75         String validEmail = "tholmquist_gj@yahoo.com";
76         CSCD437RegExMethods.emailPattern(validEmail, new PrintStream(testOut));
77         assertTrue(testOut.toString().contains("NOT"));
78     }
79
80     @Test
81     public void emailPattern_invalid_format_prints_invalid_response4() {
82         String validEmail = "tholmquist@gmail.com";
83         CSCD437RegExMethods.emailPattern(validEmail, new PrintStream(testOut));
84         assertTrue(testOut.toString().contains("NOT"));
85     }
86
87
88 }
```

Name:

The screenshot shows two instances of the IntelliJ IDEA IDE running side-by-side. Both instances have the same project structure and code content, indicating they are likely screenshots of the same session.

Project Structure:

- TL-Lab5
- sources root, ~/Desktop/Programs/CSCD4
- .idea
- out
- CSCD437Lab5RegexTests
- CSCD437RegExMain
- CSCD437RegExMethods
- TL-Lab5.iml
- External Libraries
- Scratches and Consoles

Code Editor Content (Top Window):

```
TL-Lab5 CSCI437Lab5RegexTests.java CSCI437RegExMain.java CSCI437RegExMethods.java
19
20     @Test
21     public void name_valid_format_prints_valid_response() {
22         String validName = "Holmquist, Tyler, B";
23         CSCD437RegExMethods.namePattern(validName, new PrintStream(testOut));
24         assertFalse(testOut.toString().contains("NOT"));
25     }
26
27     @Test
28     public void name_valid_format_prints_valid_response2() {
29         String validName = "Holmquist-Brown, Tyler, B";
30         CSCD437RegExMethods.namePattern(validName, new PrintStream(testOut));
31         assertFalse(testOut.toString().contains("NOT"));
32     }
33
34     @Test
35     public void name_valid_format_prints_valid_response3() {
36         String validName = "Holmquist, Tyler, BD";
37         CSCD437RegExMethods.namePattern(validName, new PrintStream(testOut));
38         assertFalse(testOut.toString().contains("NOT"));
39 }
```

Run History (Bottom Window):

- Tests passed: 8 of 8 tests - 15ms
- CSCI437Lab5RegexTests 15ms /Library/Java/JavaVirtualMachines/jdk-11.0.16.1.jdk/Contents/Home/bin/java ...
 - name_valid_format_prints_va 1ms
 - name_valid_format_prints_val 1ms
 - name_valid_format_prints_val 1ms
 - nameValidFormatPrints_val 1ms
 - namePattern_invalid_format_p 1ms
 - namePattern_invalid_format_prints
 - namePattern_invalid_format_prints
 - namePattern_invalid_format_prints
- Process finished with exit code 0

Bottom Status Bar:

72:72 LF UTF-8 4 spaces

The screenshot shows the IntelliJ IDEA interface with the project 'TL-Lab5' open. The 'CSCD437Lab5RegexTests.java' file is the active editor, displaying three test methods for the `namePattern` function. The 'Run' tool window at the bottom shows a successful run of the tests, with 8 tests passed in 15ms. The terminal output indicates a process finished with exit code 0.

```

57     @Test
58     public void namePattern_invalid_format_prints_invalid_response2() {
59         String validName = "Holmquist-Tyler-B";
60         CSCD437RegExMethods.namePattern(validName, new PrintStream(testOut));
61         assertTrue(testOut.toString().contains("NOT"));
62     }
63
64     @Test
65     public void namePattern_invalid_format_prints_invalid_response3() {
66         String validName = "Holmquist.Tyler.B";
67         CSCD437RegExMethods.namePattern(validName, new PrintStream(testOut));
68         assertTrue(testOut.toString().contains("NOT"));
69     }
70
71     @Test
72     public void namePattern_invalid_format_prints_invalid_response4() {
73         String validName = "Holmquist-Brown Tyler B";
74         CSCD437RegExMethods.namePattern(validName, new PrintStream(testOut));
75         assertTrue(testOut.toString().contains("NOT"));
76     }
77

```

Date:

For this method I could not figure out how to properly include february 29th on leap years

The screenshot shows the IntelliJ IDEA interface with the project 'TL-Lab5' open. The 'CSCD437Lab5RegexTests.java' file is the active editor, displaying three test methods for the `datePattern` function. The 'Run' tool window at the bottom shows a successful run of the tests, with 17 tests passed in 20ms. The terminal output indicates a process finished with exit code 0.

```

18
19
20     @Test
21     public void datePattern_valid_format_prints_valid_response() {
22         String validDate = "09-08-2003";
23         CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
24         assertFalse(testOut.toString().contains("NOT"));
25     }
26
27     @Test
28     public void datePattern_valid_format_prints_valid_response2() {
29         String validDate = "11-23-1999";
30         CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
31         assertFalse(testOut.toString().contains("NOT"));
32     }
33
34     @Test
35     public void datePattern_valid_format_prints_valid_response3() {
36         String validDate = "09/08/2003";
37         CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
38         assertFalse(testOut.toString().contains("NOT"));
39     }
40

```

TL-Lab5 CSCD437Lab5RegexTests

Project .idea out CSCD437Lab5RegexTests CSCD437RegExMain CSCD437RegExMethods TL-Lab5.iml External Libraries Scratches and Consoles

```
41     @Test
42     public void datePattern_valid_format_prints_valid_response4() {
43         String validDate = "11/24/1987";
44         CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
45         assertFalse(testOut.toString().contains("NOT"));
46     }
47
48     @Test
49     public void datePattern_valid_format_prints_valid_response5() {
50         String validDate = "3/23/2018";
51         CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
52         assertFalse(testOut.toString().contains("NOT"));
53     }
54
55     @Test
56     public void datePattern_valid_format_prints_valid_response6() {
57         String validDate = "10/24/1989";
58         CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
59         assertFalse(testOut.toString().contains("NOT"));
60     }
```

Run: CSCD437Lab5RegexTests Tests passed: 17 of 17 tests – 20 ms

CSCD437Lab5RegexTests 20ms /Library/Java/JavaVirtualMachines/jdk-11.0.16.1.jdk/Contents/Home/bin/java ...

- datePattern_valid_format_pri 2ms
- datePattern_valid_format_prints_v4
- datePattern_valid_format_prin 1ms
- datePattern_valid_format_prin 2ms
- datePattern_valid_format_prin 1ms
- datePattern_valid_format_prin 1ms
- datePattern_valid_format_prin 1ms
- datePattern_valid_format_prints_v6
- datePattern_invalid_format_prints_j
- datePattern_invalid_format_prints_j
- datePattern_invalid_format_prints_j
- datePattern_invalid_format_prints_j

Process finished with exit code 0

81:5 LF UTF-8 4 spaces

TL-Lab5 CSCD437Lab5RegexTests

Project .idea out CSCD437Lab5RegexTests CSCD437RegExMain CSCD437RegExMethods TL-Lab5.iml External Libraries Scratches and Consoles

```
61     @Test
62     public void datePattern_valid_format_prints_valid_response7() {
63         String validDate = "11-30-1999";
64         CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
65         assertFalse(testOut.toString().contains("NOT"));
66     }
67
68     @Test
69     public void datePattern_valid_format_prints_valid_response8() {
70         String validDate = "3-4-2020";
71         CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
72         assertFalse(testOut.toString().contains("NOT"));
73     }
74
75     @Test
76     public void datePattern_valid_format_prints_valid_response9() {
77         String validDate = "2/28/2019";
78         CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
79         assertFalse(testOut.toString().contains("NOT"));
80     }
```

Run: CSCD437Lab5RegexTests Tests passed: 17 of 17 tests – 20 ms

CSCD437Lab5RegexTests 20ms /Library/Java/JavaVirtualMachines/jdk-11.0.16.1.jdk/Contents/Home/bin/java ...

- datePattern_valid_format_pri 2ms
- datePattern_valid_format_prints_v6
- datePattern_valid_format_prin 1ms
- datePattern_valid_format_prin 2ms
- datePattern_valid_format_prin 1ms
- datePattern_valid_format_prin 1ms
- datePattern_valid_format_prin 1ms
- datePattern_valid_format_prints_v8
- datePattern_invalid_format_prints_j
- datePattern_invalid_format_prints_j
- datePattern_invalid_format_prints_j
- datePattern_invalid_format_prints_j

Process finished with exit code 0

81:5 LF UTF-8 4 spaces

TL-Lab5 > CSCD437Lab5RegexTests

Project > TL-Lab5 sources root, ~/Desktop/Programs/CSCD4 > .idea > out > CSCD437Lab5RegexTests > CSCD437RegExMain > CSCD437RegExMethods > TL-Lab5.iml > External Libraries > Scratches and Consoles

```
86     @Test
87     public void datePattern_invalid_format_prints_invalid_response() {
88         String validDate = "10 24 2013";
89         CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
90         assertTrue(testOut.toString().contains("NOT"));
91     }
92
93     @Test
94     public void datePattern_invalid_format_prints_invalid_response2() {
95         String validDate = "05/22/2020";
96         CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
97         assertTrue(testOut.toString().contains("NOT"));
98     }
99
100    @Test
101    public void datePattern_invalid_format_prints_invalid_response3() {
102        String validDate = "03-14/2015";
103        CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
104        assertTrue(testOut.toString().contains("NOT"));
105    }
106
```

Run: CSCD437Lab5RegexTests > Tests passed: 17 of 17 tests - 20 ms

CSCD437Lab5RegexTests 20ms /Library/Java/JavaVirtualMachines/jdk-11.0.16.1.jdk/Contents/Home/bin/java ...

- datePattern_valid_format_pri 2ms
- datePattern_valid_format_prints_v4
- datePattern_valid_format_prin 1ms
- datePattern_valid_format_prin 2ms
- datePattern_valid_format_prin 1ms
- datePattern_invalid_format_prints_j
- datePattern_invalid_format_prints_j
- datePattern_invalid_format_prints_j
- datePattern_invalid_format_prints_j

Process finished with exit code 0

81:5 LF UTF-8 4 spaces

TL-Lab5 > CSCD437Lab5RegexTests

Project > TL-Lab5 sources root, ~/Desktop/Programs/CSCD4 > .idea > out > CSCD437Lab5RegexTests > CSCD437RegExMain > CSCD437RegExMethods > TL-Lab5.iml > External Libraries > Scratches and Consoles

```
107     @Test
108     public void datePattern_invalid_format_prints_invalid_response4() {
109         String validDate = "18/12/2020";
110         CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
111         assertTrue(testOut.toString().contains("NOT"));
112     }
113
114     @Test
115     public void datePattern_invalid_format_prints_invalid_response5() {
116         String validDate = "10/33/2019";
117         CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
118         assertTrue(testOut.toString().contains("NOT"));
119     }
120
121     @Test
122     public void datePattern_invalid_format_prints_invalid_response6() {
123         String validDate = "3-35-1989";
124         CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
125         assertTrue(testOut.toString().contains("NOT"));
126     }
127
```

Run: CSCD437Lab5RegexTests > Tests passed: 17 of 17 tests - 20 ms

CSCD437Lab5RegexTests 20ms /Library/Java/JavaVirtualMachines/jdk-11.0.16.1.jdk/Contents/Home/bin/java ...

- datePattern_valid_format_pri 2ms
- datePattern_valid_format_prints_v4
- datePattern_valid_format_prin 1ms
- datePattern_valid_format_prin 2ms
- datePattern_valid_format_prin 1ms
- datePattern_invalid_format_prints_j
- datePattern_invalid_format_prints_j
- datePattern_invalid_format_prints_j
- datePattern_invalid_format_prints_j

Process finished with exit code 0

81:5 LF UTF-8 4 spaces

Military Time:

```
TL-Lab5 > CSCD437Lab5RegexTests
```

```
Project > TL-Lab5 sources root, ~/Desktop/Programs/CSCD4
```

```
CSCD437Lab5RegexTests.java
```

```
CSCD437RegExMain.java
```

```
CSCD437RegExMethods.java
```

```
TL-Lab5.iml
```

```
External Libraries
```

```
Scratches and Consoles
```

```
127
    @Test
    public void datePattern_invalid_format_prints_invalid_response7() {
        String validDate = "2/30/1925";
        CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
        assertTrue(testOut.toString().contains("NOT"));
    }

    @Test
    public void datePattern_invalid_format_prints_invalid_response8() {
        String validDate = "2/29/2019";
        CSCD437RegExMethods.datePattern(validDate, new PrintStream(testOut));
        assertTrue(testOut.toString().contains("NOT"));
    }
```

```
Run: CSCD437Lab5RegexTests > Tests passed: 17 of 17 tests - 20 ms
```

```
Process finished with exit code 0
```

```
TL-Lab5 > CSCD437Lab5RegexTests
```

```
Project > TL-Lab5 sources root, ~/Desktop/Programs/CSCD4
```

```
CSCD437Lab5RegexTests.java
```

```
CSCD437RegExMain.java
```

```
CSCD437RegExMethods.java
```

```
TL-Lab5.iml
```

```
External Libraries
```

```
Scratches and Consoles
```

```
19
    @Test
    public void militaryTime_valid_format_prints_valid_response() {
        String validTime = "23:15:34";
        CSCD437RegExMethods.militaryTimePattern(validTime, new PrintStream(testOut));
        assertFalse(testOut.toString().contains("NOT"));
    }

    @Test
    public void militaryTime_valid_format_prints_valid_response2() {
        String validTime = "09:24:59";
        CSCD437RegExMethods.militaryTimePattern(validTime, new PrintStream(testOut));
        assertFalse(testOut.toString().contains("NOT"));
    }

    @Test
    public void militaryTime_valid_format_prints_valid_response3() {
        String validTime = "09:04:25";
        CSCD437RegExMethods.militaryTimePattern(validTime, new PrintStream(testOut));
        assertFalse(testOut.toString().contains("NOT"));
    }
```

```
Run: CSCD437Lab5RegexTests > Tests passed: 8 of 8 tests - 15ms
```

```
Process finished with exit code 0
```

```
TL-Lab5 > CSCD437Lab5RegexTests
```

```
Project > TL-Lab5 sources root, ~/Desktop/Programs/CSCD4
```

```
CSCD437Lab5RegexTests.java
```

```
CSCD437RegExMain.java
```

```
CSCD437RegExMethods.java
```

```
TL-Lab5.iml
```

```
External Libraries
```

```
Scratches and Consoles
```

```
15ms
    militaryTime_valid_format_pri_2ms
    militaryTime_valid_format_prin1ms
    militaryTime_valid_format_prints_v1
    militaryTimePattern_invalid_for1ms
    militaryTimePattern_invalid_for1ms
    militaryTimePattern_invalid_for1ms
    militaryTimePattern_invalid_for1ms
    militaryTime_valid_format_prints_v1
```

```
Run: CSCD437Lab5RegexTests > Tests passed: 8 (moments ago)
```

Military Time:

```
TL-Lab5 > CSCD437Lab5RegexTests
```

```
Project > TL-Lab5 sources root, ~/Desktop/Programs/CSCD4
```

```
CSCD437Lab5RegexTests.java
```

```
CSCD437RegExMain.java
```

```
CSCD437RegExMethods.java
```

```
TL-Lab5.iml
```

```
External Libraries
```

```
Scratches and Consoles
```

```
19
    @Test
    public void militaryTime_valid_format_prints_valid_response() {
        String validTime = "23:15:34";
        CSCD437RegExMethods.militaryTimePattern(validTime, new PrintStream(testOut));
        assertFalse(testOut.toString().contains("NOT"));
    }

    @Test
    public void militaryTime_valid_format_prints_valid_response2() {
        String validTime = "09:24:59";
        CSCD437RegExMethods.militaryTimePattern(validTime, new PrintStream(testOut));
        assertFalse(testOut.toString().contains("NOT"));
    }

    @Test
    public void militaryTime_valid_format_prints_valid_response3() {
        String validTime = "09:04:25";
        CSCD437RegExMethods.militaryTimePattern(validTime, new PrintStream(testOut));
        assertFalse(testOut.toString().contains("NOT"));
    }
```

```
Run: CSCD437Lab5RegexTests > Tests passed: 8 of 8 tests - 15ms
```

```
Process finished with exit code 0
```

```
TL-Lab5 > CSCD437Lab5RegexTests
```

```
Project > TL-Lab5 sources root, ~/Desktop/Programs/CSCD4
```

```
CSCD437Lab5RegexTests.java
```

```
CSCD437RegExMain.java
```

```
CSCD437RegExMethods.java
```

```
TL-Lab5.iml
```

```
External Libraries
```

```
Scratches and Consoles
```

```
15ms
    militaryTime_valid_format_pri_2ms
    militaryTime_valid_format_prin1ms
    militaryTime_valid_format_prints_v1
    militaryTimePattern_invalid_for1ms
    militaryTimePattern_invalid_for1ms
    militaryTimePattern_invalid_for1ms
    militaryTimePattern_invalid_for1ms
    militaryTime_valid_format_prints_v1
```

```
Run: CSCD437Lab5RegexTests > Tests passed: 8 (moments ago)
```

TL-Lab5 > CSCD437Lab5RegexTests > militaryTimePattern_invalid_format_prints_inval...

Project > TL-Lab5 sources root, ~/Desktop/Programs/CSCD4

out

CSCD437Lab5RegexTests

CSCD437RegExMain

CSCD437RegExMethods

TL-Lab5.iml

External Libraries

Scratches and Consoles

```
38     }
39
40     @Test
41     public void militaryTime_valid_format_prints_valid_response4() {
42         String validTime = "14:48:18";
43         CSCD437RegExMethods.militaryTimePattern(validTime, new PrintStream(testOut));
44         assertFalse(testOut.toString().contains("NOT"));
45     }
46
47
48
49
50     @Test
51     public void militaryTimePattern_invalid_format_prints_invalid_response() {
52         String validTime = "24:00:00";
53         CSCD437RegExMethods.militaryTimePattern(validTime, new PrintStream(testOut));
54         assertTrue(testOut.toString().contains("NOT"));
55     }
56
57     @Test
58     public void militaryTimePattern_invalid_format_prints_invalid_response2() {
```

Run: CSCD437Lab5RegexTests >

Tests passed: 8 of 8 tests - 15ms

CSCD437Lab5RegexTests 15ms /Library/Java/JavaVirtualMachines/jdk-11.0.16.1.jdk/Contents/Home/bin/java ...

militaryTime_valid_format_pri 12ms

militaryTime_valid_format_prin 1ms

militaryTime_valid_format_prints_v 1ms

militaryTimePattern_invalid_for 1ms

militaryTimePattern_invalid_for 1ms

militaryTimePattern_invalid_for 1ms

militaryTimePattern_invalid_for 1ms

militaryTime_valid_format_v 1ms

Process finished with exit code 0

73:37 LF UTF-8 4 spaces

TL-Lab5 > CSCD437Lab5RegexTests > militaryTimePattern_invalid_format_prints_inval...

Project > TL-Lab5 sources root, ~/Desktop/Programs/CSCD4

out

CSCD437Lab5RegexTests

CSCD437RegExMain

CSCD437RegExMethods

TL-Lab5.iml

External Libraries

Scratches and Consoles

```
57     }
58     @Test
59     public void militaryTimePattern_invalid_format_prints_invalid_response2() {
60         String validTime = "22:60:00";
61         CSCD437RegExMethods.militaryTimePattern(validTime, new PrintStream(testOut));
62         assertTrue(testOut.toString().contains("NOT"));
63     }
64
65     @Test
66     public void militaryTimePattern_invalid_format_prints_invalid_response3() {
67         String validTime = "11:12:60";
68         CSCD437RegExMethods.militaryTimePattern(validTime, new PrintStream(testOut));
69         assertTrue(testOut.toString().contains("NOT"));
70     }
71
72     @Test
73     public void militaryTimePattern_invalid_format_prints_invalid_response4() {
74         String validTime = "22 12 15";
75         CSCD437RegExMethods.militaryTimePattern(validTime, new PrintStream(testOut));
76         assertTrue(testOut.toString().contains("NOT"));
77     }
```

Run: CSCD437Lab5RegexTests >

Tests passed: 8 of 8 tests - 15ms

CSCD437Lab5RegexTests 15ms /Library/Java/JavaVirtualMachines/jdk-11.0.16.1.jdk/Contents/Home/bin/java ...

militaryTime_valid_format_pri 12ms

militaryTime_valid_format_prin 1ms

militaryTime_valid_format_prints_v 1ms

militaryTimePattern_invalid_for 1ms

militaryTimePattern_invalid_for 1ms

militaryTimePattern_invalid_for 1ms

militaryTimePattern_invalid_for 1ms

militaryTime_valid_format_v 1ms

Process finished with exit code 0

68:56 LF UTF-8 4 spaces

Money Pattern:

The screenshots show the IntelliJ IDEA interface with two code editors and their corresponding run outputs.

Top Screenshot:

- Code Editor:** CSDC437Lab5RegexTests.java
- Content:** Test cases for valid money patterns.
- Run Output:** Tests passed: 8 of 8 tests - 22ms
- Test Cases:**
 - @Test public void moneyPattern_invalid_format_prints_invalid_response() { ... }
 - @Test public void moneyPattern_valid_format_prints_valid_response2() { ... }
 - @Test public void moneyPattern_valid_format_prints_valid_response3() { ... }

Bottom Screenshot:

- Code Editor:** CSDC437Lab5RegexTests.java
- Content:** Additional test cases for money patterns, including one for invalid format.
- Run Output:** Tests passed: 8 of 8 tests - 22ms
- Test Cases:**
 - @Test public void moneyPattern_valid_format_prints_valid_response4() { ... }
 - @Test public void moneyPattern_valid_format_prints_valid_response5() { ... }
 - @Test public void moneyPattern_invalid_format_prints_invalid_response() { ... }

moneyPattern_invalid_format_prints_invalid_response

```

56
57     @Test
58     public void moneyPattern_invalid_format_prints_invalid_response() {
59         String validPattern = "25.00";
60         CSCD437RegExMethods.moneyPattern(validPattern, new PrintStream(testOut));
61         assertTrue(testOut.toString().contains("NOT"));
62     }
63
64     @Test
65     public void moneyPattern_invalid_format_prints_invalid_response2() {
66         String validPattern = "$10000.00";
67         CSCD437RegExMethods.moneyPattern(validPattern, new PrintStream(testOut));
68         assertTrue(testOut.toString().contains("NOT"));
69     }
70
71     @Test
72     public void moneyPattern_invalid_format_prints_invalid_response3() {
73         String validPattern = "$87";
74         CSCD437RegExMethods.moneyPattern(validPattern, new PrintStream(testOut));
75         assertTrue(testOut.toString().contains("NOT"));
76     }

```

Run: CSCD437Lab5RegexTests

Tests passed: 8 of 8 tests – 22 ms

- CSCD437Lab5RegexTests 22 ms
 - ✓ moneyPattern_invalid_format 8ms
 - ✓ moneyPattern_invalid_format_print 8ms
 - ✓ moneyPattern_valid_format_p 1ms
 - ✓ moneyPattern_invalid_format_print 1ms
 - ✓ moneyPattern_valid_format_prints 1ms
 - ✓ moneyPattern_valid_format_p 1ms
 - ✓ moneyPattern_valid_format_p 1ms
 - ✓ moneyPattern_valid_format_p 1ms

Process finished with exit code 0

Password Pattern:

passwordPattern_valid_format_prints_valid_response

```

19
20     @Test
21     public void passwordPattern_valid_format_prints_valid_response() {
22         String validPassword = "WhyNotTry!13";
23         CSCD437RegExMethods.passwordPattern(validPassword, new PrintStream(testOut));
24         assertFalse(testOut.toString().contains("NOT"));
25     }
26
27     @Test
28     public void passwordPattern_valid_format_prints_valid_response2() {
29         String validPassword = "YouDontKnow12?";
30         CSCD437RegExMethods.passwordPattern(validPassword, new PrintStream(testOut));
31         assertFalse(testOut.toString().contains("NOT"));
32     }
33
34     @Test
35     public void passwordPattern_valid_format_prints_valid_response3() {
36         String validPassword = "GeeSeLayEggs34Q";
37         CSCD437RegExMethods.passwordPattern(validPassword, new PrintStream(testOut));
38         assertFalse(testOut.toString().contains("NOT"));
39     }

```

Run: CSCD437Lab5RegexTests

Tests passed: 6 of 6 tests – 13 ms

- CSCD437Lab5RegexTests 13 ms
 - ✓ passwordPattern_valid_form 11ms
 - ✓ passwordPattern_valid_format_prin 1ms
 - ✓ passwordPattern_valid_format1ms
 - ✓ passwordPattern_invalid_form 1ms
 - ✓ passwordPattern_invalid_format_pr 1ms
 - ✓ passwordPattern_invalid_format_pr 1ms

Process finished with exit code 0

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project View:** Shows the project structure under "TL-Lab5".
- Code Editor:** Displays the file `CSCD437Lab5RegexTests.java` containing Java test methods for password validation.
- Run Tab:** Shows the output of the run command, indicating 6 tests passed in 13ms.
- Bottom Status Bar:** Shows "Tests passed: 6 (moments ago)" and other system information like file size and encoding.

```
40     @Test
41     public void passwordPattern_invalid_format_prints_invalid_response() {
42         String validPassword = "IdontKnowYou!";
43         CSCD437RegExMethods.passwordPattern(validPassword, new PrintStream(testOut));
44         assertTrue(testOut.toString().contains("NOT"));
45     }
46
47     @Test
48     public void passwordPattern_invalid_format_prints_invalid_response2() {
49         String validPassword = "GeeseLayEggs348";
50         CSCD437RegExMethods.passwordPattern(validPassword, new PrintStream(testOut));
51         assertTrue(testOut.toString().contains("NOT"));
52     }
53
54     @Test
55     public void passwordPattern_invalid_format_prints_invalid_response3() {
56         String validPassword = "WhyNotTry?";
57         CSCD437RegExMethods.passwordPattern(validPassword, new PrintStream(testOut));
58         assertTrue(testOut.toString().contains("NOT"));
59     }
```

Run: CSCD437Lab5RegexTests ×
Tests passed: 6 of 6 tests – 13ms
Process finished with exit code 0

Tests passed: 6 (moments ago)