|  |  |  |
| --- | --- | --- |
| ID / Bezeichnung | 1 | Ist Samstag Test |
| Beschreibung | Mit einem künftigen oder vergangenen Datum, welches ein Samstag ist wir getestet. | |
| Testvoraussetzung | Ein DateTime Objekt ist instanziiert und die isSaturday Methode wir aufgerufen. | |
| Testschritte  (Schritt / erwartetes Resultat) | Laden der Umgebungsvariablen, ein WorkDay Object mit einem Samstag Datum erstellen und das WorkDay Objekt fragen ob ein Samstag gesetzt ist. | |
| Erwartetes Ergebnis | OK | |
| Erhaltenes Ergebnis | OK | |

|  |  |
| --- | --- |
| Tester | Lars Gächter |
| Datum Testdurchführung | 27.05.2021 |
| Fehlerbeschreibung | Keine |

Code:

*/\*\*  
 \** ***@throws*** *\Symfony\Contracts\HttpClient\Exception\TransportExceptionInterface  
 \*/*public function testIsSaturday(): void  
{  
 (new Dotenv())->bootEnv(dirname(*\_\_DIR\_\_*).'\..\..\.env');  
 $dateTime = RIODateTimeFactory::*getDateTime*("2021-05-22");  
 $workDay = new RIOWorkDayObject($dateTime);  
 $result = $workDay->isSaturday();  
 $this->assertEquals(true,$result,"Test has failed!");  
}

|  |  |  |
| --- | --- | --- |
| ID / Bezeichnung | 2 | Ist Sonntag Test |
| Beschreibung | Mit einem künftigen oder vergangenen Datum, welches ein Sonntag ist wir getestet. | |
| Testvoraussetzung | Ein DateTime Objekt ist instanziiert und die isSunday Methode wir aufgerufen. | |
| Testschritte  (Schritt / erwartetes Resultat) | Laden der Umgebungsvariablen, ein WorkDay Object mit einem Sonntag Datum erstellen und das WorkDay Objekt fragen ob ein Sonntag gesetzt ist. | |
| Erwartetes Ergebnis | OK | |
| Erhaltenes Ergebnis | OK | |

|  |  |
| --- | --- |
| Tester | Lars Gächter |
| Datum Testdurchführung | 27.05.2021 |
| Fehlerbeschreibung | Keine |

Code:

*/\*\*  
 \** ***@throws*** *\Symfony\Contracts\HttpClient\Exception\TransportExceptionInterface  
 \*/*public function testIsSunday(): void  
{  
 (new Dotenv())->bootEnv(dirname(*\_\_DIR\_\_*).'\..\..\.env');  
 $dateTime = RIODateTimeFactory::*getDateTime*("2021-05-23");  
 $workDay = new RIOWorkDayObject($dateTime);  
 $result = $workDay->isSunday();  
 $this->assertEquals(true,$result,"Test has failed!");  
}

|  |  |  |
| --- | --- | --- |
| ID / Bezeichnung | 3 | Ist Feiertag Test |
| Beschreibung | Mit einem künftigen oder vergangenen Datum, welches ein Feiertag ist wir getestet. | |
| Testvoraussetzung | Ein DateTime Objekt ist instanziiert und die isRIOHoliday Methode wir aufgerufen. | |
| Testschritte  (Schritt / erwartetes Resultat) | Laden der Umgebungsvariablen, ein WorkDay Object mit einem Feiertag Datum erstellen und das WorkDay Objekt fragen ob ein Feiertag gesetzt ist. | |
| Erwartetes Ergebnis | OK | |
| Erhaltenes Ergebnis | OK | |

|  |  |
| --- | --- |
| Tester | Lars Gächter |
| Datum Testdurchführung | 27.05.2021 |
| Fehlerbeschreibung | Keine |

Code:

*/\*\*  
 \* Test  
 \*  
 \** ***@throws*** *\PHPUnit\Framework\ExpectationFailedException|\Exception  
 \** ***@throws*** *\Symfony\Contracts\HttpClient\Exception\TransportExceptionInterface  
 \*/*public function testIsHoliday(): void  
{  
 (new Dotenv())->bootEnv(dirname(*\_\_DIR\_\_*).'\..\..\.env');  
 $dateTime = RIODateTimeFactory::*getDateTime*("2021-05-24");  
 $workDay = new RIOWorkDayObject($dateTime);  
 $result = $workDay->isRIOHoliday();  
 $this->assertEquals(true,$result,"Test has failed!");  
}

|  |  |  |
| --- | --- | --- |
| ID / Bezeichnung | 4 | Ist heute Pflichtzeit Test |
| Beschreibung | Mit dem heutigen Datum wird getestet.  Ist der Test rot ist keine Pflichtzeit, ist der Test grün gilt eine Pflichtzeit. | |
| Testvoraussetzung | Ein DateTime Objekt ist instanziiert und die isRIOHoliday oder isSunday oder isSaturday Methoden werden aufgerufen. | |
| Testschritte  (Schritt / erwartetes Resultat) | Laden der Umgebungsvariablen, ein WorkDay Object mit dem heutigen Datum (siehe Datum Testdurchführung) erstellen und das WorkDay Objekt fragen ob ein Feiertag, ein Sonntag oder ein Samstag ist gesetzt ist. | |
| Erwartetes Ergebnis | OK | |
| Erhaltenes Ergebnis | OK | |

|  |  |
| --- | --- |
| Tester | Lars Gächter |
| Datum Testdurchführung | 27.05.2021 |
| Fehlerbeschreibung | Keine |

Code:

*/\*\*  
 \** ***@throws*** *\Symfony\Contracts\HttpClient\Exception\TransportExceptionInterface  
 \*/*public function testMandatoryTimeToday(): void  
{  
 (new Dotenv())->bootEnv(dirname(*\_\_DIR\_\_*).'\..\..\.env');  
 $workDay = new RIOWorkDayObject();  
 $result = $workDay->isRIOHoliday() ||  
 $workDay->isSunday() ||  
 $workDay->isSaturday();  
 $this->assertEquals(false,$result,"Test has failed!");  
}

Meldung der Resultate:

Testing started at 14:43 ...

PHPUnit 9.5.4 by Sebastian Bergmann and contributors.

Time: 00:00.476, Memory: 6.00 MB

OK (4 tests, 4 assertions)

Process finished with exit code 0