|  |  |  |  |
| --- | --- | --- | --- |
| **Java class/method being tested: Data.java/isValid() method** | | | |
| **Test Case** | **Requirement** | **Test Description and Input Data** | **Expected Result/Output** |
| 1 | The valid range for the month shall be 1-12 | * Create an instance of Date with valid date and year, but the month is 0. * Test data: “0/12/2022” | False |
| 2 | The valid range for the month shall be 1-12 | * Create an instance of Date with valid date and year, but the month is greater than 12. * Test data: “13/12/2022” | False |
| 3 | The valid range for the day shall be 1-31 for the longer months | * Create an instance of Date with a valid long month and year, but the day is greater than 31. * Test data: “1/32/2022” | False |
| 4 | The valid range for the day shall be 1-31 for the longer months | * Create an instance of Date with valid long month and year, and the date is 31 (edge case). * Test data: “7/31/2022” | True |
| 5 | The valid range for the day in February shall be 1-28 on a non-leap year | * Create an instance of Date with month 2 and valid year, and the date is 29. * Test data: “2/29/2022” | False |
| 6 | The valid range for the day in February shall be 1-28 on a non-leap year | * Create an instance of Date with month 2 and valid year, and the date is 28. * Test data: “2/28/2022” | True |
| 7 | February 29th shall only be valid on a leap year. | * Create an instance of Date with month 2, day 29, and a valid leap year. * Test data: “2/29/2020” | True |
| 8 | February 29th shall only be valid on a leap year. | * Create an instance of Date with month 2, day 29, and a non-leap year. * Test data: “2/29/2100” | False |
| 9 | February 29th shall only be valid on a leap year. | * Create an instance of Date with month 2, day 29, and a valid leap year. * Test data: “2/29/2000” | True |
| 10 | The valid range for the short non-February months shall be 1-30 | * Create an instance of Date with valid short month and year, but the day is greater than 30. * Test data: “4/31/2022” | False |
| 11 | The year shall not be negative | * Create an instance of Date with valid date and month, but the year is negative. * Test data: “1/12/-2012” | False |

|  |  |  |  |
| --- | --- | --- | --- |
| **Java class/method being tested: Timeslot.java/compareTo() method** | | | |
| **Test Case** | **Requirement** | **Test Description and Input Data** | **Expected Result/Output** |
| 1 | If the calling timeslot has a later date than the parameter slot, output must be 1 | * Create two Timeslot objects with one date later than the other * Calling timeslot: “02/06/2022 9:15” * Parameter timeslot: “01/12/2022 10:00” | 1 |
| 2 | If the calling timeslot has an earlier date than the parameter slot, output must be -1 | * Create two Timeslot objects with one date earlier than the other * Calling timeslot: “12/17/2021 9:15” * Parameter timeslot: “03/21/2022 10:00” | -1 |
| 3 | If the calling and parameter timeslot dates are equal, then if the calling timeslot has a later time than the parameter slot, output must be 1 | * Create two Timeslot objects with one date later than the other * Calling timeslot: “06/24/2022 11:15” * Parameter timeslot: “06/24/2022 10:30” | 1 |
| 4 | If the calling and parameter timeslot dates are equal, then if the calling timeslot has an earlier time than the parameter slot, output must be -1 | * Create two Timeslot objects with one date later than the other * Calling timeslot: “04/18/2022 11:15” * Parameter timeslot: “04/18/2022 14:45” | -1 |
| 5 | If the calling and parameter timeslots have the same date and time, output must be 0 | * Create two Timeslot objects with one date later than the other * Calling timeslot: “07/04/2022 15:30” * Parameter timeslot: “07/04/2022 15:30” | 0 |