Author Robert Bonagura

Test cases for Date class.

All object names in the Expected Output column are in reference to the objects names used to Date.main()

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | Description | Input  (Method being tested, followed by the parameters it receives as input) | Expected Output  (The return value of each method tested in the previous column) |
| 1 | Tests default constructor and checks that isValid() returns true on valid dates. | Constructor: “1/01/2000”  isValid(): dates[0] | dates[0]  true |
| 2 | Tests default constructor and checks that isValid() returns false on invalid dates. | Constructor: “13/01/2002”  isValid(): dates[3] | dates[3]  false |
| 3 | Tests default constructor and checks that isValid() returns true or false appropriately when isLeapYear() is called. | Case a:  Constructor: “02/28/2007”  IsLeapYear(): dates[6]  IsValid(): dates[6]  Case b:  Constructor: “02/29/2002”  IsLeapYear(): dates[8]  IsValid(): dates[8]  Case c:  Constructor: “02/29/2000”  IsLeapYear(): dates[7]  IsValid(): dates[7] | Case a:  dates[6]  false  true  Case b:  dates[8]  false  false  Case c:  dates[7]  true  true |
| 4 | Test copy constructor and that equals() method returns true and false appropriately. | Case a:  Copy Constructor: date1  date1.equals(): date2  Case b:  Copy Constructor: dates[0]  Copy Constructor: dates[1]  date1.equals(): date4 | Case a:  date2  true  Case b:  date1  date4  false |
| 5 | Test toString() method. Date.main() prints date objects using System.out.printf(), which implicitly calls the date object’s toString() method. | Case a:  System.out.printf():  “%s equals %s: %s\n”, date1, date2, date1.equals(date2)  Case b:  System.out.printf():  “%s equals %s: %s\n”, date1, date4, date1.equals(date4) | Case a:  “1/1/2000 equals 1/1/2000: true”  Case b:  “1/1/2000 equals 6/1/2007: false” |

Author Ezra Haleva

Test cases for TeamMember class.

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | Description | Sample Input | Expected result / output |
| 1 | Test the constructor and check if the name provided is equal to the name variable in the resulting object. Equivalence is determined using the String.equals() method. | “Name1” | The name variable of the resultant object is equivalent to the string “Name1”. |
| 2 | Test the constructor and check if the date variable in the resulting object is equal to a date object with the same values. Uses the Date.equals() method to determine equivalence. | “1/1/2020” | The startDate variable of the resultant object is equivalent to a new Date variable constructed with the string “1/1/2020” |
| 3 | Tests getStartDate() method and checks if the returned date object is equivalent to the startDate variable. Uses the Date.equals() method to determine equivalence. | The object has been initialized with a start date. | The Date variable returned by the getStartDate() method is equivalent to the startDate variable of the object. |
| 4 | Tests getName() method and checks if returned string is equivalent to the name variable of the object. Uses String.equals() method to determine equivalence. | The object has been initialized with a name. | The string value returned by the getName() method is equivalent to the name variable of the object. |
| 5 | Tests the equals() method and checks if it’s returning the correct equivalence values. Uses constructor to create TeamMember objects with the corresponding values. | Case 1:  (“TestName1” ,“1/1/2020”) compared with  (“TestName1”, “1/1/2020”)  Case 2:  (“TestName1”, “1/1/2020”) compared with (“TestName1”, “2/2/2020”)  Case 3:  (“TestName1”, “1/1/2020”) compared with (“TestName2”, “1/1/2020”) | Case 1:  equals method returns true.  Case 2:  equals method returns false.  Case 3:  equals method returns false. |
| 6 | Tests the toString method and checks if the String output is equal to the expected string output given the known values of name and startDate. Uses the constructor to create object and uses Date.toString to produce string representation of date. Uses String.equals() to determine equivalence. | The object has been initialized with name “Name1” and startDate = newDate(“1/1/2020”) | toString() method returns a string = “Name1 1/1/2020” |