| **class name:** Date  **Method signature:** public boolean isValid() {} // Check if a given date is valid | | | |
| --- | --- | --- | --- |
| **Test Case #** | **Requirement** | **Test Description and Input Data** | **Expected result/output** |
| 1 | Number of days in February for a non-  leap year shall be 28. The method shall  return false if the date given has 29  days for a non-leap year. | * Create an instance of Date with month=2 and day>28 and year is not divisible by 4 * Test data: "2/29/2011" | false |
| 2 | Number of days in February for a non-  leap year shall be 28. The method shall  return false if the date given has 29  days for a non-leap year. | * Create an instance of Date with month=2 and day<=29 and year is divisible by 4 * Test data: "2/29/2020" | true |
| 3 | Valid range for the month shall be >= 1  and <= 12. The method shall return  false for a month value outside the  valid range. | * Create an instance of Date with month<1 or month>12 and a valid day and year * Test data: “13/28/2018” | false |
| 4 | The method shall return false for any date with a year before 1900. | * Create an instance of Date with year<1900 with a valid month and day * Test data: “2/28/1899” | false |
| 5 | Number of days in February for a non-  leap year shall be 28. The method shall  return false if the date given has 29  days for a non-leap year. | * Create an instance of Date with month=2 and day<=28 and year is not divisible by 4 * Test data: "2/28/2011" | true |
| 6 | Number of days in February for a non-  leap year shall be 28. The method shall  return false if the date given has 29  days for a non-leap year. | * Create an instance of Date with month=2 and day>29 and year is divisible by 4 * Test data: "2/30/2020" | false |
| 7 | Valid range for the month shall be >= 1  and <= 12. The method shall return  false for a month value outside the  valid range. | * Create an instance of Date with 1<=month<=12 and a valid day and year * Test data: “12/28/2018” | true |
| 8 | The method shall return false for any date with a year before 1900. | * Create an instance of Date with year>=1900 with a valid month and day * Test data: “2/28/1900” | true |

| **class name:** Event  **Method signature:** public boolean equals(Object obj) {} // Compares the Date and Timeslot of two Event | | | |
| --- | --- | --- | --- |
| **Test Case #** | **Requirement** | **Test Description and Input Data** | **Expected result/output** |
| 1 | Compare two Event s’ Date and Timeslot parameters. Return true if the Date and Timeslot are exactly the same regardless of other parameters. | * Create two instances of Event with the exact same Date and Timeslot * Test data1: new Event(new Date("09/28/2023"), Timeslot.MORNING, Location.AB2225, new Contact(Department.CS.getName(), "ph390@scarletmail.rutgers.edu"), 90); * Test data2: new Event(new Date("09/28/2023"), Timeslot.MORNING, Location.AB2225, new Contact(Department.CS.getName(), "ph390@scarletmail.rutgers.edu"), 90); | true |
| 2 | Compare two Event s’ Date and Timeslot parameters. Return true if the Date and Timeslot are exactly the same regardless of other parameters. | * Create two instances of Event with the different Date but same Timeslot * Test data1: new Event(new Date("09/28/2023"), Timeslot.MORNING, Location.AB2225, new Contact(Department.CS.getName(), "ph390@scarletmail.rutgers.edu"), 90); * Test data2: new Event(new Date("09/28/2021"), Timeslot.MORNING, Location.AB2225, new Contact(Department.CS.getName(), "ph390@scarletmail.rutgers.edu"), 90); | false |
| 3 | Compare two Event s’ Date and Timeslot parameters. Return true if the Date and Timeslot are exactly the same regardless of other parameters. | * Create two instances of Event with the same Date but different Timeslot * Test data1: new Event(new Date("09/28/2023"), Timeslot.MORNING, Location.AB2225, new Contact(Department.CS.getName(), "ph390@scarletmail.rutgers.edu"), 90); * Test data2: new Event(new Date("09/28/2023"), Timeslot.EVENING, Location.AB2225, new Contact(Department.CS.getName(), "ph390@scarletmail.rutgers.edu"), 90); | false |
| 4 | Compare two Event s’ Date and Timeslot parameters. Return true if the Date and Timeslot are exactly the same regardless of other parameters. | * Create two instances of Event with the exact same Date and Timeslot but different Location * Test data1: new Event(new Date("09/28/2023"), Timeslot.MORNING, Location.AB2225, new Contact(Department.CS.getName(), "ph390@scarletmail.rutgers.edu"), 90); * Test data2: new Event(new Date("09/28/2023"), Timeslot.MORNING, Location.MU302, new Contact(Department.CS.getName(), "ph390@scarletmail.rutgers.edu"), 90); | true |