Java Class name: **Date** **Method Signature:** **public boolean isValid()**			
Test Case #	Test Objective	Test Description and Input Data	Expected Result
1	The method should read a date before 1900 as invalid	Create an instance of Date with a valid day and month but with an invalid year < 1900. Test input: “8/16/1880”	invalid
2	The method should read a month value greater than 12 as invalid	Create a Date instance with a valid year and day, but a month > 12. Test input: “27/8/2002”	invalid
3	The method should read a day value greater than 31 for january as invalid	Create a date instance with the month 1 and a valid year, but with the day > 31 Test input: “1/40/2011”	invalid
4	The method should read a day value of 29 for the month of february in the year 2017 as invalid, as 2017 is not a leap year	Create a date instance with the month 2, day 29, but the year a non-leap year. Test Input: “2/29/2017”	invalid
5	The method should accept a day value of 29 for the month of february in the year 2016, as it is a leap year.	Create a date instance with the day 29, month 2, and year 2016. Test Input: “2/29/2016”	valid
6	The method should accept a date of March 28, 1968 as a valid date	Create a date instance with the day 28, month 3, and year 1968 Test Input:	valid

		“3/28/1968”	

Java Class name: Profile **Method Signature: public int compareTo()**			
Test Case #	Test Objective	Test Description and Input Data	Expected Result
1	Verify comparison when this profile's last name comes alphabetically before the other's.	Compare two profiles with different last names where the first profile's last name precedes the second's. **Input Data:** ● Profile 1: new Profile("John", "Apple", new Date(2000, 1, 1)) ● Profile 2: new Profile("John", "Banana", new Date(2000, 1, 1))	-1
2	Verify comparison when last names match but this profile's first name comes first alphabetically.	Compare two profiles with identical last names but differing first names. **Input Data:** ● Profile 1: new Profile("Alice", "Doe", new Date(1995, 5, 15)) ● Profile 2: new Profile("Bob", "Doe", new Date(1995, 5, 15))	-1
3	Verify comparison when names are identical but this profile has an earlier birth date.	Compare two profiles with identical names but different birth dates. **Input Data:** ● Profile 1: new	-1

		Profile("John", "Doe", new Date(1990, 1, 1)) ● Profile 2: new Profile("John", "Doe", new Date(2000, 1, 1))	
4	Verify comparison when this profile's last name comes alphabetically after the other's.	Compare two profiles with different last names where the first profile's last name follows the second's. **Input Data:** ● Profile 1: new Profile("John", "Zebra", new Date(2000, 1, 1)) ● Profile 2: new Profile("John", "Apple", new Date(2000, 1, 1))	1
5	Verify comparison when last names match but this profile's first name comes later alphabetically.	● Compare two profiles with identical last names but differing first names. **Input Data:** ● Profile 1: new Profile("Zack", "Smith", new Date(1985, 3, 10)) ● Profile 2: new Profile("Alice", "Smith", new Date(1985, 3, 10))	1
6	Verify comparison when names are identical but this profile	Compare two profiles with identical names but different birth dates.	1

	has a later birth date.	**Input Data:** ● Profile 1: new Profile("Jane", "Doe", new Date(2005, 12, 31)) ● Profile 2: new Profile("Jane", "Doe", new Date(2000, 5, 20))	
7	Verify comparison when all fields are identical.	Compare two profiles with the same first name, last name, and date of birth. **Input Data:** ● Profile 1: new Profile("John", "Doe", new Date(1995, 5, 15)) ● Profile 2: new Profile("John", "Doe", new Date(1995, 5, 15))	0