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	A date of birth in January must be at most 31 days	-Create an instance of the Date class with a valid day in January -test input: “1/31/2003”	True
2	A date of birth in January must be at most 31 days	-Create an instance of the Date class with an invalid day in January -test input: “1/32/2003”	False
3	A date of birth in February must be at most 29 days on a leap year	-Create an instance of the Date class with a valid day in February on a leap year -test input: “2/29/2004”	True
4	A date of birth in February must be at most 29 days on a leap year	-Create an instance of the Date class with an invalid day in February on a leap year -test input: “2/31/2004”	False
5	A date of birth in February must be at most 28 days on a non-leap year	-Create an instance of the Date class with a valid day in February on a non-leap year -test input: “2/28/2003”	True
6	A date of birth in February must be at most 28 days on a non-leap year	-Create an instance of the Date class with an invalid day in February on a non-leap year	False

		-test input: “2/29/2003”	
7	A date of birth in March must be at most 31 days	-Create an instance of the Date class with a valid day in March -test input: “4/31/2003”	True
8	A date of birth in March must be at most 31 days	-Create an instance of the Date class with an invalid day in March -test input: “4/32/2003”	False
9	A date of birth in April must be at most 30 days	-Create an instance of the Date class with a valid day in April -test input: “5/30/2003”	True
10	A date of birth in April must be at most 30 days	-Create an instance of the Date class with an invalid day in April -test input: “5/32/2003”	False
11	A date of birth in June must be at most 30 days	-Create an instance of the Date class with a valid day in June -test input: “6/30/2003”	True
12	A date of birth in June must be at most 30 days	-Create an instance of the Date class with an invalid day in June -test input: “6/32/2003”	False
13	A date of birth in July must be at most 31 days	-Create an instance of the Date class with a valid day in July -test input: “7/31/2003”	True
14	A date of birth in July must be at most 31 days	-Create an instance of the Date class with an invalid day in July -test input: “7/32/2003”	False

15	A date of birth in August must be at most 31 days	-Create an instance of the Date class with a valid day in August -test input: “8/31/2003”	True
16	A date of birth in August must be at most 31 days	-Create an instance of the Date class with an invalid day in August -test input: “8/32/2003”	False
17	A date of birth in September must be at most 30 days	-Create an instance of the Date class with a valid day in September -test input: “9/30/2003”	True
18	A date of birth in September must be at most 30 days	-Create an instance of the Date class with an invalid day in September -test input: “9/32/2003”	False
19	A date of birth in October must be at most 31 days	-Create an instance of the Date class with a valid day in October -test input: “9/32/2003”	True
20	A date of birth in October must be at most 31 days	-Create an instance of the Date class with an invalid day in October -test input: “10/31/2003”	False
21	A date of birth in November must be at most 30 days	-Create an instance of the Date class with a valid day in November -test input: “11/30/2003”	True
22	A date of birth in November must be at most 30 days	-Create an instance of the Date class with an invalid day in	False

		November -test input: “11/32/2003”	
23	A date of birth in December must be at most 31 days	-Create an instance of the Date class with a valid day in December -test input: “12/31/2003”	True
24	A date of birth in December must be at most 31 days	-Create an instance of the Date class with an invalid day in December -test input: “12/32/2003”	False
25	Valid range of months is 1-12	Create an instance of the date class with a month of over 12 -test input: “13/31/2003”	True
26	Valid range of months is 1-12	Create an instance of the date class with a month under 1 -test input “0/1/2003	False
27	The day of the month will never be under 1	Create an instance of the date class with a day under 1 -test input: “1/0/2003”	False
28	The date cannot be in the future	Create an instance of the date class that is past today’s date -test input: “1/1/2222”	False
29	A date of birth in May must be at most 31 days	Create an instance of the Date cass with a valid day in May -test input: “3/31/2003”	True
30	A date of birth in Ma/y must be at most 31 days	Create an instance of the Date cass with a valid day in May -test input: “3/32/2003	False

Class name: Student Method Signature: public void compareTo(Student s){} //returns 0 if the students are equal, otherwise returns a negative value if the students' names or DOB is above student s, and a positive value if the opposite is true.			
Case Number	Condition	Test Input	Expected Output
1	Returns 0 if 2 students are equal	Create 2 identical instances of a student and compare them. Test Input: " -John Smith 1/1/2002 -John Smith 1/1/2002 "	0
2	Returns a negative number if one student is lower than the main student.	Create 2 differeng students and compare them to eachother Test Input: " -John Smith 1/1/2002 -Alex Jones 1/1/2002	-9
3	Returns a positive number if one student is higher lexographically than the main student.	Create 2 differing students and compare them to eachother. Test input: " -Alex Jones 1/1/2002 -John Smith 1/1/2002	9
4	Returns a positive number if the compared student has an identical name but is younger.	Create 2 students with the same first and last names but different birthdays Test input: " -John Smith 1/1/2002 -John Smith 1/2/2002	-1
5	Returns a negative number if the compared student has an identical name but is older	Create 2 students with the same first and last names but different birthdays Test Input: " -John Smith 1/2/2002 -John Smith 1/1/2002	1