

Java class/method being tested: addMember method			
Test Case #	Requirement	Test Description and Input Data	Expected result/output
1	First name cannot be empty	<p>The first name will be an empty input value with remaining values being valid.</p> <p>First name: Last name: Brown Date of Birth: 10/7/1991 Location: Piscataway Member type: Standard</p>	"Enter a first name!"
2	First name cannot contain anything other than characters	<p>The first name will be an integer with remaining values being valid.</p> <p>First name: 123 Last name: Brown Date of Birth: 10/7/1991 Location: Piscataway Member type: Standard</p>	"Only letters allowed in first name!"
3	Last name cannot be empty	<p>The last name will be an empty input value with remaining values being valid.</p> <p>First name: Carl Last name: Date of Birth: 10/7/1991 Location: Piscataway Member type: Standard</p>	"Enter a last name!"
4	Last name cannot contain anything other than characters	<p>The last name will be an integer with remaining values being valid.</p> <p>First name: Carl Last name: 123 Date of Birth: 10/7/1991 Location: Piscataway Member type: Standard</p>	"Only letters allowed in last name!"

5	User did not pick a date	<p>There will be no date selected from the date menu.</p> <p>First name: Carl Last name: Brown Date of Birth: Location: Piscataway Member type: Standard</p>	“Enter a date!”
6	User did not pick a location	<p>There will be no location selected from the location menu.</p> <p>First name: Carl Last name: Brown Date of Birth: 10/7/1991 Location: Member type: Standard</p>	“Choose a location!”
7	User is already in the database	<p>The info inputted will be inputted a second time after the member is already added.</p> <p>First name: Carl Last name: Brown Date of Birth: 10/31/2000 Location: Piscataway Member type: Standard</p>	“Carl Brown is already in the database.”
8	User is not 18 or older	<p>The date inputted will show that the user is less than 18 years of age.</p> <p>First name: Carl Last name: Brown Date of Birth: 10/31/2010 Location: Piscataway Member type: Standard</p>	“DOB 10/31/2010: must be 18 or older to join!”
9	User inputs a date that is in the future.	<p>The date inputted will be one that is in the future.</p> <p>First name: Carl Last name: Brown Date of Birth:</p>	“DOB 10/31/2023: cannot be today or a future date!”

		10/31/2023 Location: Piscataway Member type: Standard	
10	User inputs correct info and adds a member to the database.	The member info exists in the database and is successfully added to the database. First name: Carl Last name: Brown Date of Birth: 10/31/2000 Location: Piscataway Member type: Standard	"Carl Brown added."

Java class/method being tested: removeMember method			
Test Case #	Requirement	Test Description and Input Data	Expected result/output
1	First name cannot be empty	The first name will be an empty input value with remaining values being valid. First name: Last name: Brown Date of Birth: 10/7/1991 Location: Piscataway Member type: Standard	"Enter a first name!"
2	First name cannot contain anything other than characters	The first name will be an integer with remaining values being valid. First name: 123 Last name: Brown Date of Birth: 10/7/1991 Location: Piscataway Member type: Standard	"Only letters allowed in first name!"
3	Last name cannot be empty	The last name will be an empty input value with remaining values being	"Enter a last name!"

		<p>valid.</p> <p>First name: Carl Last name: Date of Birth: 10/7/1991 Location: Piscataway Member type: Standard</p>	
4	Last name cannot contain anything other than characters	<p>The last name will be an integer with remaining values being valid.</p> <p>First name: Carl Last name: 123 Date of Birth: 10/7/1991 Location: Piscataway Member type: Standard</p>	“Only letters allowed in last name!”
5	User has already been removed or any information does not match up with member info in the database.	<p>The member is already removed and the user tries to remove the member with the same exact info.</p> <p>First name: Carl Last name: Brown Date of Birth: 10/31/2000 Location: Piscataway Member type: Standard</p>	“Carl Brown is not in the database.”
6	User inputs correct info and removes a member from the database.	<p>The member info exists in the database and is successfully removed from the database.</p> <p>First name: Carl Last name: Brown Date of Birth: 10/31/2000 Location: Piscataway Member type: Standard</p>	“Carl Brown removed.”

Java class/method being tested: checkIn method

Test Case #	Requirement	Test Description and Input Data	Expected result/output
1	Member does not exist	<p>The member that is being checked in does not exist in the member database</p> <p>Class: Cardio Instructor: Emma Location: Edison First name: Carl Last name: Black Date of Birth: 10/31/2000</p>	"Carl Black 10/31/2000 is not in the database."
2	Member is already in the class	<p>The member that is being checked in is already checked into the class.</p> <p>Class: Cardio Instructor: Emma Location: Edison First name: Carl Last name: Brown Date of Birth: 10/31/2000</p>	"Carl Brown already checked in."
3	Member with standard membership is checking in at invalid location	<p>The member that is being checked in has a standard membership while trying to check in at a different location.</p> <p>Class: Spinning Instructor: Emma Location: Franklin First name: Carl Last name: Brown Date of Birth: 10/31/2000</p>	"Carl Brown checking in FRANKLIN, 08873, SOMERSET - standard membership location restriction.!"
4	Class with the selected teacher at selected location does not exist	<p>The class that the member is checking into does not exist.</p> <p>Class: Spinning Instructor: Emma Location: Bridgewater First name: Carl</p>	"SPINNING by Emma does not exist at BRIDGEWATER, 08807, SOMERSET."

		Last name: Brown Date of Birth: 10/31/2000	
5	User is trying to check in a guest with a standard membership.	The member has a standard membership and tries to check in a guest. Class: Cardio Instructor: Emma Location: Bridgewater First name: Carl Last name: Brown Date of Birth: 10/31/2000	“Standard membership - guest check-in is not allowed.”
6	User is checking in a guest without available guest passes.	The member has a family membership and is trying to check in a guest after running out of guest passes. Class: Cardio Instructor: Emma Location: Edison First name: Carl Last name: Black Date of Birth: 10/31/2000	“Carl Black ran out of guest pass.”
7	User does not select an instructor	There is no instructor chosen. Class: Cardio Instructor: Location: Edison First name: Carl Last name: Black Date of Birth: 10/31/2000	“Choose an instructor!”
8	User does not select a location	There is no location chosen to check into. Class: Cardio Instructor: Emma Location: First name: Carl Last name: Brown Date of Birth:	“Choose a location!”

		10/31/2000	
9	User does not enter any characters for their first name.	<p>The member that is being checked in does not have a first name.</p> <p>Class: Cardio Instructor: Emma Location: Edison First name: Last name: Brown Date of Birth: 10/31/2000</p>	“Enter a first name!”
10	User does not enter any characters for their last name.	<p>The member that is being checked in does not have a last name.</p> <p>Class: Cardio Instructor: Emma Location: Edison First name: Carl Last name: Date of Birth: 10/31/2000</p>	“Enter a last name!”
11	User does not enter a date of birth	<p>The member that is being checked in does not have a date of birth.</p> <p>Class: Cardio Instructor: Emma Location: Edison First name: Carl Last name: Brown Date of Birth:</p>	“Enter a date!”
12	User enters something other than characters for their first name.	<p>The member that is being checked in does not input letters for their first name.</p> <p>Class: Cardio Instructor: Emma Location: Edison First name: 123 Last name: Brown Date of Birth: 10/31/2000</p>	“Only letters allowed in first name!”
13	User enters something	The member that is	“Only letters allowed in

	other than characters for their last name.	being checked in does not input letters for their last name. Class: Cardio Instructor: Emma Location: Edison First name: Carl Last name: 123 Date of Birth: 10/31/2000	last name!"
14	User enters correct info and member is checked into the class.	The member that is being checked in inputs the right info and has not already checked in. Class: Cardio Instructor: Emma Location: Edison First name: Carl Last name: Brown Date of Birth: 10/31/2000	"Carl Brown checked in. CARDIO - EMMA, 18:30, EDISON - Participants - Carl Brown, DOB: 10/31/2000, Membership expires 2/7/2023, Location: EDISON, 08837, MIDDLESEX, (Family) guest-pass remaining: 1"
15	User enters correct info and guest is checked into the class. A guest pass is removed	The guest that is being checked in inputs the right info and has not already checked in. Class: Cardio Instructor: Emma Location: Edison First name: Carl Last name: Brown Date of Birth: 10/31/2000	"Chris Tai (guest) checked in. CARDIO - DAVIS, 18:30, BRIDGEWATER - Guests - Chris Tai, DOB: 11/6/2002, Membership expires 11/7/2023, Location: BRIDGEWATER, 08807, SOMERSET, (Premium) Guest-pass remaining: 2"

Java class/method being tested: checkOut method			
Test Case #	Requirement	Test Description and Input Data	Expected result/output
1	Member does not exist	The member that is	"Carl Black 10/31/2000

		<p>being checked out does not exist in the member database</p> <p>Class: Cardio Instructor: Emma Location: Edison First name: Carl Last name: Black Date of Birth: 10/31/2000</p>	is not in the database.”
2	Member has not checked into the class	<p>The member that is being checked out has not checked into the class.</p> <p>Class: Cardio Instructor: Emma Location: Edison First name: Carl Last name: Brown Date of Birth: 10/31/2000</p>	“Carl Brown did not check in.”
3	Class with the selected teacher at selected location does not exist	<p>The class that the member is checking out of does not exist.</p> <p>Class: Spinning Instructor: Emma Location: Bridgewater First name: Carl Last name: Brown Date of Birth: 10/31/2000</p>	“SPINNING by Emma does not exist at BRIDGEWATER, 08807, SOMERSET.”
4	User does not select an instructor	<p>There is no instructor chosen.</p> <p>Class: Cardio Instructor: Location: Edison First name: Carl Last name: Black Date of Birth: 10/31/2000</p>	“Choose an instructor!”
5	User does not select a location	<p>There is no location chosen to check out of.</p>	“Choose a location!”

		Class: Cardio Instructor: Emma Location: First name: Carl Last name: Brown Date of Birth: 10/31/2000	
6	User does not enter any characters for their first name.	The member that is being checked out does not have a first name. Class: Cardio Instructor: Emma Location: Edison First name: Last name: Brown Date of Birth: 10/31/2000	"Enter a first name!"
7	User does not enter any characters for their last name.	The member that is being checked out does not have a last name. Class: Cardio Instructor: Emma Location: Edison First name: Carl Last name: Date of Birth: 10/31/2000	"Enter a last name!"
8	User does not enter a date of birth	The member that is being checked out does not have a date of birth. Class: Cardio Instructor: Emma Location: Edison First name: Carl Last name: Brown Date of Birth:	"Enter a date!"
9	User enters something other than characters for their first name.	The member that is being checked out does not input letters for their first name. Class: Cardio Instructor: Emma Location: Edison	"Only letters allowed in first name!"

		First name: 123 Last name: Brown Date of Birth: 10/31/2000	
10	User enters something other than characters for their last name.	The member that is being checked out does not input letters for their last name. Class: Cardio Instructor: Emma Location: Edison First name: Carl Last name: 123 Date of Birth: 10/31/2000	"Only letters allowed in last name!"
11	User info is correct and the member was checked into the class.	The member being checked out was checked in and inputs valid data. Class: Cardio Instructor: Emma Location: Edison First name: Carl Last name: Brown Date of Birth: 10/31/2000	"Carl Brown done with the class."
12	User info is correct and the guest was checked into the class.	The guest being checked out was checked in and inputs valid data. Class: Cardio Instructor: Emma Location: Edison First name: Carl Last name: Brown Date of Birth: 10/31/2000	Carl Brown Guest done with the class.

Java class/method being tested: print method			
Test Case #	Requirement	Test Description and Input Data	Expected result/output

1	No members in the database.	There are no members registered in the database. Input: print()	"Member database is empty!"
2	Standard member has been loaded	One member (John Doe) has been loaded, and is to be printed out Input: print()	-list of members- John Doe, DOB: 1/20/2004, Membership expires 2/15/2023, Location: BRIDGEWATER, 08807, SOMERSET -end of list-
3	Family member has been loaded	One family member has been loaded, and is to be printed out Input: print()	-list of members- Chris Tai, DOB: 11/5/2003, Membership expires 2/7/2023, Location: FRANKLIN, 08873, SOMERSET, (Family) guest-pass remaining: 1 -end of list-
4	Premium member has been loaded	One family member has been loaded, and is to be printed out Input: print()	-list of members- Chris Tai, DOB: 11/5/2003, Membership expires 2/7/2023, Location: FRANKLIN, 08873, SOMERSET, (Premium) guest-pass remaining: 3 -end of list-

Java class/method being tested: printByName method			
Test Case #	Requirement	Test Description and Input Data	Expected result/output
1	No members in the database.	There are no members registered in the database. Input: printByName()	"Member database is empty!"
2	The member with the earlier last name should be printed out	Two members with different last names (John Doe and Duke	-list of members- John Doe, DOB: 1/20/2003, Membership

		Ellington) have been loaded, and are to be printed out Input: printByName()	expired 3/30/2021, Location: BRIDGEWATER, 08807, SOMERSET Duke Ellington, DOB: 2/29/2000, Membership expires 4/30/2023, Location: FRANKLIN, 08873, SOMERSET -end of members-
3	The member with the earlier first name should be printed out (if they have the same last name)	Two members with different the same first names but different last names (John Doe and Jane Doe) have been loaded, and are to be printed out Input: printByName()	Jane Doe, DOB: 5/1/1996, Membership expires 6/20/2023, Location: EDISON, 08837, MIDDLESEX John Doe, DOB: 1/20/2004, Membership expires 2/15/2023, Location: BRIDGEWATER, 08807, SOMERSET

Java class/method being tested: printByExpirationDate			
Test Case #	Requirement	Test Description and Input Data	Expected result/output
1	If there are no members in the database, the result will reflect that	There are no members registered in the database. Input: printByDate()	"Member database is empty!"
2	The member with the earlier expiration date should be printed first	Two members with different expiration dates have been loaded, and are to be printed out (Jane Doe with expiration date 6/20/2023 and Kate Lindsey with expiration date 8/31/2023) Input: printByDate()	Jane Doe, DOB: 5/1/1996, Membership expires 6/20/2023, Location: EDISON, 08837, MIDDLESEX Kate Lindsey, DOB: 7/15/1977, Membership expires 8/31/2023, Location: SOMERVILLE, 08876, SOMERSET

Java class/method being tested: printByCounty			
Test Case #	Requirement	Test Description and Input Data	Expected result/output
1	If there are no members in the database, the result will reflect that	There are no members registered in the database. Input: printByCounty()	"Member database is empty!"
2	The member from the county that comes earlier in the alphabet will be printed first	Two members from different counties have been loaded, and are to be printed out (April March from Middlesex and John Doe from Bridgewater) Input: printByCounty()	April March, DOB: 3/31/1990, Membership expires 12/30/2022, Location: PISCATAWAY, 08854, MIDDLESEX John Doe, DOB: 1/20/2004, Membership expires 2/15/2023, Location: BRIDGEWATER, 08807, SOMERSET
3	The member from the town that comes earlier in the alphabet will be printed first	Two members from the same county, but different towns have been loaded, and are to be printed out Input: printByCounty()	Duke Ellington, DOB: 2/29/2000, Membership expires 4/30/2023, Location: FRANKLIN, 08873, SOMERSET Kate Lindsey, DOB: 7/15/1977, Membership expires 8/31/2023, Location: SOMERVILLE, 08876, SOMERSET

Java class/method being tested: printWithMembershipFees			
Test Case #	Requirement	Test Description and Input Data	Expected result/output
1	If there are no members in the database, the result will reflect that	There are no members registered in the database. Input: printWithMembershipFees()	"Member database is empty!"

2	Standard member is printed out with a standard fee	A standard member has been loaded Input: printWithMembershipFee()	Roy Brooks, DOB: 8/8/1977, Membership expired 9/30/2020, Location: SOMERVILLE, 08876, SOMERSET, Membership fee: \$149.96
3	Family member is printed out with a family fee	A family member has been loaded Input: printWithMembershipFee()	Christopher Tai, DOB: 6/10/2003, Membership expires 11/7/2023, Location: BRIDGEWATER, 08807, SOMERSET, (Family) guest-pass remaining: 1, Membership fee: \$209.96
3	Premium member is printed out with a family fee	A premium member has been loaded Input: printWithMembershipFee()	Christopher Tai, DOB: 6/10/2003, Membership expires 11/7/2023, Location: BRIDGEWATER, 08807, SOMERSET, (Premium) guest-pass remaining: 3, Membership fee: \$659.89

Java class/method being tested: loadSchedule and loadMember method			
Test Case #	Requirement	Test Description and Input Data	Expected result/output
1	Classes to be loaded has already been loaded.	The file that is loading classes has already been loaded. Input: Duplicate class.txt file	“-Fitness Classes loaded- Pilates already loaded. Spinning already loaded. Cardio already loaded. Pilates already loaded. Spinning already loaded. Cardio already loaded. Pilates already loaded. Pilates already loaded. Cardio already loaded. Spinning already loaded. Cardio already loaded. Spinning already loaded. Spinning already loaded. Pilates already loaded. Cardio already loaded. -end of class list-”
2	Classes are not loaded yet	Classes in the file have not yet been loaded in Input: Load class.txt file	“-Fitness Classes loaded- PILATES - JENNIFER, 9:30, BRIDGEWATER SPINNING - JENNIFER, 14:00, PISCATAWAY CARDIO - JENNIFER, 18:30,

			<p>SOMERVILLE PILATES - KIM, 9:30, FRANKLIN SPINNING - KIM, 14:00, FRANKLIN CARDIO - KIM, 18:30, PISCATAWAY PILATES - DAVIS, 9:30, PISCATAWAY PILATES - DAVIS, 14:00, EDISON CARDIO - DAVIS, 18:30, BRIDGEWATER SPINNING - DENISE, 9:30, BRIDGEWATER CARDIO - DENISE, 14:00, SOMERVILLE SPINNING - DENISE, 18:30, EDISON SPINNING - EMMA, 9:30, FRANKLIN PILATES - EMMA, 14:00, EDISON CARDIO - EMMA, 18:30, EDISON -end of class list-”</p>
3	Members to be loaded has already been loaded	<p>The file that is loading members has already been loaded.</p> <p>Input: Duplicate members.txt file</p>	<p>“-list of members loaded- John Doe, DOB: 1/20/2004, Membership expires 2/15/2023, Location: BRIDGEWATER, 08807, SOMERSET John Doe, DOB: 1/20/2003, Membership expired 3/30/2021, Location: BRIDGEWATER, 08807, SOMERSET Jane Doe, DOB: 5/1/1996, Membership expires 6/20/2023, Location: EDISON, 08837, MIDDLESEX April March, DOB: 3/31/1990, Membership expires 12/30/2022, Location: PISCATAWAY, 08854, MIDDLESEX Mary Lindsey, DOB: 12/1/1989, Membership expires 5/31/2023, Location: FRANKLIN, 08873, SOMERSET Duke Ellington, DOB: 2/29/2000, Membership expires 4/30/2023, Location: FRANKLIN, 08873, SOMERSET Roy Brooks, DOB: 8/8/1977, Membership expired 9/30/2020, Location: SOMERVILLE, 08876, SOMERSET Roy Brooks, DOB: 9/9/1977, Membership expired 10/30/2022, Location: SOMERVILLE, 08876, SOMERSET Kate Lindsey, DOB: 7/15/1977, Membership expires 8/31/2023, Location: SOMERVILLE, 08876, SOMERSET Carl Brown, DOB: 10/7/1991, Membership expires 3/31/2023, Location: PISCATAWAY, 08854, MIDDLESEX Paul Siegel, DOB: 5/1/1999, Membership expired 1/31/2022, Location: EDISON,</p>

			08837, MIDDLESEX Bill Scanlan, DOB: 5/1/1999, Membership expires 1/31/2023, Location: EDISON, 08837, MIDDLESEX -end of list-
4	Members are not loaded yet	Members in the file have not yet been loaded in Input: Load class.txt file	“-list of members loaded- John Doe already in database John Doe already in database Jane Doe already in database April March already in database Mary Lindsey already in database Duke Ellington already in database Roy Brooks already in database Roy Brooks already in database Kate Lindsey already in database Carl Brown already in database Paul Siegel already in database Bill Scanlan already in database -end of list-