Tab: <u>Open/Close</u>			
Test Case #	**Requirement**	**Test Description and Input Data**	**Expected Outputs/Results**
1	Invalid first name given to open or close an account by leaving it blank	➔ Leaving “Enter First Name” field empty ➔ And entering proper data for all the other fields	“Please enter a first name”
2	Valid first name given to open an account by entering a value	➔ Entering “John” for first name field ➔ And entering proper data for all other fields ➔ Account type can be Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/12/2003 ([Account Type]) opened.”
3	Valid first name given to close an account by entering a value	➔ Entering “John” for first name field ➔ And entering proper data for all other fields ➔ Account type can be Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/12/2003 ([Account Type]) has been closed.”
4	Invalid last name given to open or close an account by leaving it blank	➔ Leaving “Enter Last Name” field empty ➔ And entering proper data for all the other fields	“Please enter a last name”
5	Valid last name given to open an account by entering a value	➔ Entering “Doe” for last name field ➔ And entering proper data for all other fields ➔ Account type can be either Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/12/2003 ([Account Type]) opened.”
6	Valid last name given to close an account by entering a value	➔ Entering “Doe” for last name field ➔ And entering proper data for all other fields ➔ Account type can be either Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/12/2003 ([Account Type]) has been closed.”

Test Case #	Requirement	Test Description and Input Data	Expected Outputs/Results
7	Invalid date of birth where the person is under the age of 16	➔ Younger than 16 years old ➔ Entered “11/12/2023” in DOB	“DOB invalid: 11/12/2023 under 16”
8	Valid date of birth where they are over the age of 16 to open an account	➔ Older than or equal to 16 years old ➔ Entered “11/12/2003” in DOB ➔ Account type can be either Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/12/2003 ([Account Type]) opened.”
9	Valid date of birth where they are over the age of 16 to close an account	➔ Older than or equal to 16 years old ➔ Entered “11/12/2003” in DOB ➔ Account type can be either Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/12/2003 ([Account Type]) has been closed.”
10	Invalid initial deposit	➔ Either non-integer, non-double, “0” or a negative number is entered for initial deposit ➔ Entered: “xy”, “0”, “-200”	“Initial deposit cannot be 0 or negative.”
11	Valid initial deposit for Checking (C), Savings (S), College Checking (CC)	➔ Positive integer or positive double is entered ➔ Entered: “500”, “650.80” ➔ Account type can be either Checking (C), Savings (S), College Checking (CC)	“John Doe 11/12/2003 ([Account Type]) has been opened.”
12	Valid initial deposit for Money Market (MM)	➔ Positive integer or positive double is entered greater than 2000 ➔ Entered: “2500”, “2650.80”	“John Doe 11/12/2003 (MM) has been opened.”
13	Invalid initial deposit for Money Market (MM)	➔ Positive integer or positive double is entered lower than 2000 ➔ Entered: “500”, “650.80”	“Minimum of $2000 to open a Money Market account.”
14	Not picking a campus location when opening a College Checking (CC) account	➔ Not selecting any of the campus locations ➔ And entering proper data for all the other fields	“Please select an campus”

Test Case #	Requirement	Test Description and Input Data	Expected Outputs/Results
15	Picking one of the three campus location when opening a College Checking (CC) account	➔ Selecting any of the three campus locations ➔ Pick between, “New Brunswick”, “Newark”, “Camden” ➔ And entering proper data for all the other fields	“John Doe 11/12/2003 (CC) has been opened.”
16	Adding a person twice for the same account type	➔ Using same name, birthdate, and account type as a person who is already in the database ➔ Account type can be either Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/12/2003 (C) is already in the database”
17	Invalid closing of an account which is not in the database	➔ Closing an account which is not in the database	“John Do 11/12/2003 (C) is not in the database.”
18	Invalid closing of an account which is in the database but different account type	➔ Closing an account which is in the database but picking a different account type ➔ Account type can be either Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/12/2003 (S) is not in the database.”
19	Valid closing of an account which is in the database	➔ Closing an account which is in the database and has the same account type ➔ Account type can be either Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/12/2003 (C) has been closed.”
20	Trying to open a College Checking (CC) account if the person already has a Checking (C) account	➔ Trying to create a College Checking account for a person who already has a checking account	“John Doe 11/12/2003 (CC) is already in the database”
21	Invalid Money Market account creation by putting initial deposit lower than $2000	➔ Trying to open a Money Market account with a initial deposit lower than $2000	“Minimum of $2000 to open a Money Market account.”

Test Case #	Requirement	Test Description and Input Data	Expected Outputs/Results
22	Valid creation of an account	➔ Giving proper values for all the fields ➔ Account type can be either Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/12/2003 ([Account Type]) opened.”
23	Trying to open a College Checking account with age 24 and above	➔ Attempting to open a College Checking account with a max age of 24 ➔ Testing age greater than 24	“DOB invalid: 11/12/1998 over 24”
24	DOB cannot be today’s date or future date	➔ Attempting to open an account with today’s date and a future date ➔ Tested: “11/6/2023”, “11/12/2029”	“DOB invalid: 11/7/2023 cannot be today or a future day.”

Tab: <u>Transactions</u>			
Test Case #	**Requirement**	**Test Description and Input Data**	**Expected Outputs/Results**
1	Invalid first name given to deposit or withdraw from an account by leaving it blank	➔ Leaving “Enter First Name” field empty ➔ And entering proper data for all the other fields	“Please enter a first name”
2	Valid first name given to deposit an account by entering a value but person is not in the database	➔ Trying to deposit a money in an account that was never opened or picking a different account type ➔ Account type can be either Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/12/2003 (C) is not in the database.”

Test Case #	Requirement	Test Description and Input Data	Expected Outputs/Results
3	Valid first name given to deposit an account by entering a value	➔ Entering “John” for first name field ➔ And entering proper data for all other fields ➔ Account type can be Checking (C), Savings (S), College Checking (CC), Money Market (MM) but has to be the same as the one picked when created	“John Doe 11/12/2003 (C) Deposit - balance updated.”
4	Valid first name given to withdraw an account by entering a value	➔ Entering “John” for first name field ➔ And entering proper data for all other fields ➔ Account type can be Checking (C), Savings (S), College Checking (CC), Money Market (MM) but has to be the same as the one picked when created	“John Doe 11/12/2003 (C) Withdraw - balance updated.”
5	Invalid last name given to deposit or withdraw from an account by leaving it blank	➔ Leaving “Enter Last Name” field empty ➔ And entering proper data for all the other fields	“Please enter a last name”
6	Valid last name given to deposit an account by entering a value but person is not in the database	➔ Trying to deposit a money in an account that was never opened or picking a different account type	“John Doe 11/12/2003 (C) is not in the database.”
7	Valid last name given to deposit an account by entering a value	➔ Entering “Doe” for last name field ➔ And entering proper data for all other fields ➔ Account type can be Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/12/2003 (C) Deposit - balance updated.”

Test Case #	Requirement	Test Description and Input Data	Expected Outputs/Results
8	Valid last name given to withdraw an account by entering a value	➔ Entering “Doe” for last name field ➔ And entering proper data for all other fields ➔ Account type can be Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/12/2003 (C) Withdraw - balance updated.”
9	Invalid date of birth where the person is under the age of 16	➔ Younger than 16 years old ➔ Entered “11/12/2023” in DOB	“DOB invalid: 11/12/2023 under 16”
10	Valid date of birth but person is not in the database	➔ Trying to deposit or withdraw with a DOB that does not match a person in the database	“John Doe 11/13/2003 (C) is not in the database.”
11	Valid date of birth where they are over the age of 16	➔ Older than or equal to 16 years old ➔ Entered “11/12/2003” in DOB ➔ Account type can be either Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/12/2003 (C) Deposit - balance updated.”
12	Valid date of birth where they are over the age of 16	➔ Older than or equal to 16 years old ➔ Entered “11/12/2003” in DOB ➔ Account type can be either Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/12/2003 (C) Withdraw - balance updated.”
13	Valid data but account type does not match anyone in database	➔ Everything besides the account type matches a person that is in the database. ➔ Account type can be either Checking (C), Savings (S), College Checking (CC), Money Market (MM)	“John Doe 11/13/2003 (MM) is not in the database.”
14	Invalid amount by leaving it blank	➔ Leaving the amount field empty and trying to deposit or withdraw ➔ All other data is entered	“Please enter an amount”

Test Case #	Requirement	Test Description and Input Data	Expected Outputs/Results
15	Entering invalid amount	➔ Non-integer, non-double ➔ Entered: “xy”	“Not a valid amount.”
16	Entering invalid numerical amount for a deposit	➔ Negative number or 0 ➔ Entered: “-500”, “0”	“Deposit - amount cannot be 0 or negative.”
17	Entering invalid numerical amount for a withdraw	➔ Negative number or 0 ➔ Entered: “-500”, “0”	“Withdraw - amount cannot be 0 or negative.”
18	Trying to withdraw more than what an account has	➔ Trying to withdraw a larger amount than what is in the account	“John Doe 11/12/2003 (C) Withdraw - insufficient fund.”
19	Valid deposit	➔ Inputting all correct data for a person that is in the database and trying to deposit funds into that account.	“John Doe 11/12/2003 (C) Deposit - balance updated.”
20	Valid withdraw	➔ Inputting all correct data for a person that is in the database and trying to withdraw funds from that account.	“John Doe 11/12/2003 (C) Withdraw - balance updated.”

Tab: <u>Database</u>			
Test Case #	**Requirement**	**Test Description and Input Data**	**Expected Outputs/Results**
1	Printing all accounts in an empty database	➔ Trying to print all the accounts without creating any accounts/on an empty database	“Account Database is empty!”

Test Case #	**Requirement**	**Test Description and Input Data**	**Expected Outputs/Results**
2	Printing all accounts in a non-empty database	➔ Printing all the accounts from a non-empty database	“*Accounts sorted by account type and profile. Checking::Jason Brown 3/31/1998::Balance $1,200.00 Checking::Kate Lindsey 8/31/2001::Balance $450.00 Money Market::Savings::Roy Brooks 10/1/2007::Balance $2,909.10::is loyal::withdrawal: 0 *end of list”
3	Updating all accounts in an empty database	➔ Trying to update all the accounts without creating any accounts/on an empty database	“Account Database is empty!”
4	Updating all accounts in a non-empty database	➔ Updating all the accounts from a non-empty database	“*list of accounts with fees and interests applied. Checking::Jason Brown 3/31/1998::Balance $1,201.00 Checking::Kate Lindsey 8/31/2001::Balance $438.38 Money Market::Savings::Roy Brooks 10/1/2007::Balance $2,920.62::is loyal::withdrawal: 0 *end of list.”
5	Printing all interests and fees in an empty database	➔ Trying to print all the interests and fees without creating any accounts/on an empty database	“Account Database is empty!”

Test Case #	**Requirement**	**Test Description and Input Data**	**Expected Outputs/Results**
6	Printing all interests and fees in a non-empty database	➔ Printing all the interests and fees from a non-empty database	“*list of accounts with fee and monthly interest Checking::Jason Brown 3/31/1998::Balance $1,201.00::fee $0.00::monthly interest $1.00 Checking::Kate Lindsey 8/31/2001::Balance $438.38::fee $12.00::monthly interest $0.37 Money Market::Savings::Roy Brooks 10/1/2007::Balance $2,920.62::is loyal::withdrawal: 0::fee $0.00::monthly interest $11.56 *end of list”
7	Uploading from a file	➔ Uploading bank accounts from a file	[List of all the accounts opened] “Jason Brown 3/31/1998(C) opened Kate Lindsey 8/31/2001(C) opened. Roy Brooks 10/1/2007(MM) opened.

Test Case #	**Requirement**	**Test Description and Input Data**	**Expected Outputs/Results**
8	Testing withdrawals on Money Market account	➔ Creating a Money Market account and making three valid withdrawals ➔ Valid withdrawals means there is sufficient balance in the account	“*Accounts sorted by account type and profile. College Checking::John Doe 11/6/1999::Balance $3,000.00::NEW_BRUNSWICK College Checking::Jason Brown 11/12/1999::Balance $3,000.00::NEW_BRUNSWICK Money Market::Savings::Roy Brooks 11/12/1998::Balance $2,700.00::is loyal::withdrawal: 3 *end of list.”
9	Withdrawals reset after updating accounts	➔ Updating accounts after withdrawals, which reset it back to zero ➔ Used data from test case 8 and updated all the accounts	“*list of accounts with fees and interests applied. College Checking::John Doe 11/6/1999::Balance $3,002.50::NEW_BRUNSWICK College Checking::Jason Brown 11/12/1999::Balance $3,002.50::NEW_BRUNSWICK Money Market::Savings::Roy Brooks 11/12/1998::Balance $2,710.69::is loyal::withdrawal: 0 *end of list.”