|  |  |  |  |
| --- | --- | --- | --- |
| Java class/method name being tested: Gym Manager GUI | | | |
| Test Case # | Requirement | Test description and Input Data | Expected result |
| 1 | Member age must be at least 18 years old. | * Enter date of birth for member under age of 18. * Test data: “9/2/2022” | “DOB 9/2/2022: must be 18 or older to join!” |
| 2 | Date of birth cannot be future date. | * Enter date of birth of future date * Test data: “12/2/2022” | “DOB 12/2/2022: cannot be today or a future birthdate!” |
| 3 | Member cannot be added to the database twice. All same data types. | * Add a member with the same first name, last name, date of birth, location, and membership type. * Test data 1:   “First Name: John  Last Name: Doe  Date of Birth: 1/20/2003  Location: Bridgewater  Membership Type: Standard”   * Test data 2:   “First Name: John  Last Name: Doe  Date of Birth: 1/20/2003  Location: Bridgewater  Membership Type: Standard” | “John Doe is already in the database.” |
| 4 | Member cannot be added to the database twice. Regardless of membership type. | * Add a member with the same first name, last name, date of birth, location, and membership type. * Test data 1:   “First Name: John  Last Name: Doe  Date of Birth: 1/20/2003  Location: Bridgewater  Membership Type: Premium”   * Test data 2:   “First Name: John  Last Name: Doe  Date of Birth: 1/20/2003  Location: Bridgewater  Membership Type: Standard” | “John Doe is already in the database.” |
| 5 | Member cannot be added to the database twice. Regardless of gym location. | * Add a member with the same first name, last name, date of birth, location, and membership type. * Test data 1:   “First Name: John  Last Name: Doe  Date of Birth: 1/20/2003  Location: Bridgewater  Membership Type: Standard”   * Test data 2:   “First Name: John  Last Name: Doe  Date of Birth: 1/20/2003  Location: Edison  Membership Type: Standard” | “John Doe is already in the database.” |
| 6 | Will not accept location that is not gym location. | * Add member with a location that doesn’t exist. * Test data 1:   “First Name: Jane  Last Name: Doe  Date of Birth: 6/1/1996  Location: ABC  Membership Type: Standard” | “ABC: invalid location!” |
| 7 | Cannot remove member twice. | * Add a member, then attempt to remove twice. * Test data 1:   “First Name: Paul  Last Name: Siegel  Date of Birth: 5/1/1999”  Click Add   * Test data 2:   “First Name: Paul  Last Name: Siegel  Date of Birth: 5/1/1999”  Click Remove Twice | “Paul Siegel is not in the database.” |
| 8 | Cannot check in for class if membership has expired. | * Load member list from “memberList.txt”. Then attempt to check a member in for a fitness class with an expired expiration date. * Test data:   “Member First Name: Roy  Member Last Name: Brooks  Member Date of Birth: 8/8/1977  Class Location: somerville  Class Name: cardio  Instructor Name: Jennifer  Member or Guest? Member”  Click Check In | “Roy Brooks 8/8/1977 membership expired.” |
| 9 | Cannot check in for fitness class if instructor doesn’t exist. | * Load member list from “memberList.txt”. Then attempt to check a member in for a fitness class with a fake instructor name. * Test data:   “Member First Name: Roy  Member Last Name: Brooks  Member Date of Birth: 8/8/1977  Class Location: somerville  Class Name: cardio  Instructor Name: Mary  Member or Guest? Member”  Click Check In | “Mary - instructor does not exist.” |
| 10 | Cannot check member into a fitness class if member doesn’t exist. | * Enter erroneous member data and attempt to check into a real fitness class. * Test data:   “Member First Name: Happy  Member Last Name: Brooks  Member Date of Birth: 8/8/1977  Class Location: somerville  Class Name: cardio  Instructor Name: Mary  Member or Guest? Member”  Click Check In | “Happy Brooks 8/8/1977 is not in the database.” |
| 11 | Cannot check member into a fitness class twice. | * Import member and class data to use to attempt to check a member into a class two times. * Test data:   “Member First Name: Mary  Member Last Name: Lindsey  Member Date of Birth: 12/1/1989  Class Location: Franklin  Class Name: Pilates  Instructor Name: Kim  Member or Guest? Member”  Click Check In two times | “Mary Lindsey checked in PILATES - KIM, 9:30, FRANKLIN  - Participants -  Mary Lindsey, DOB: 12/1/1989, Membership expires 5/31/2023, Location: FRANKLIN, 08873, SOMERSET  Mary Lindsey already checked in.” |
| 12 | Cannot check a member into a fitness class if there is a time conflict with another class. | * Import member and class data to use to check a member into one class and then attempt to check them into a class with conflict. * Test data1:   “Member First Name: Mary  Member Last Name: Lindsey  Member Date of Birth: 12/1/1989  Class Location: Franklin  Class Name: Pilates  Instructor Name: Kim  Member or Guest? Member”  Click Check In   * Test data2:   “Member First Name: Mary  Member Last Name: Lindsey  Member Date of Birth: 12/1/1989  Class Location: Franklin  Class Name: Spinning  Instructor Name: Emma  Member or Guest? Member”  Click Check In | “Mary Lindsey checked in PILATES - KIM, 9:30, FRANKLIN  - Participants -  Mary Lindsey, DOB: 12/1/1989, Membership expires 5/31/2023, Location: FRANKLIN, 08873, SOMERSET  Time conflict - SPINNING - EMMA, 9:30, FRANKLIN, 08873, SOMERSET” |
| 13 | Cannot check a member into a class that doesn’t exist. | * Import member and class data to use to attempt to check a member into a class that doesn’t exist. * Test data:   “Member First Name: Jane  Member Last Name: Doe  Member Date of Birth: 5//1/1996  Class Location: Franklin  Class Name: abc  Instructor Name: Emma  Member or Guest? Member”  Click Check In | “abc - class does not exist.” |
| 14 | Cannot check a member into a class at a location that isn’t in the system. | * Import member and class data to use to attempt to check a member into a class that’s location isn’t in the system. * Test data:   “Member First Name: Jane  Member Last Name: Doe  Member Date of Birth: 5/1/1996  Class Location: Morristown  Class Name: pilates  Instructor Name: Emma  Member or Guest? Member”  Click Check In | “Morristown - invalid location.” |
| 15 | Cannot check a member out of a class twice. | * Import member and class data to use to check a member into a class, then attempt to remove them twice. * Test data:   “Member First Name: Mary  Member Last Name: Lindsey  Member Date of Birth: 12/1/1989  Class Location: Franklin  Class Name: Pilates  Instructor Name: Kim  Member or Guest? Member”  Click Check In  Click Check Out twice | “Mary Lindsey done with the class.  Mary Lindsey did not check in.” |
| 16 | Cannot check in a guest for a member with standard membership | * Import member and class data to use to attempt to check a guest of a member of standard membership into a class. * Test data:   “Member First Name: Jane  Member Last Name: Doe  Member Date of Birth: 5/1/1996  Class Location: bridgewater  Class Name: spinning  Instructor Name: Denise  Member or Guest? Guest”  Click Check In | “Standard membership - guest check-in is not allowed.” |
| 17 | Cannot check in guest to class that is not member’s location. | * Import member and class data to use to attempt to check a guest into a class on a location that the member doesn’t belong to. * Test data:   “Member First Name: Jonnathan  Member Last Name: Wei  Member Date of Birth: 9/21/1992  Class Location: EDISON  Class Name: CARDIO  Instructor Name: EMMA  Member or Guest? Guest”  Click Check In | “Jonnathan Wei Guest checking in EDISON, 8837, MIDDLESEX - guest location restriction.” |
| 18 | Cannot check in guest to class if member has no guest passes left. | * Import member and class data to use to attempt to check a guest into a class 4 times, so the member runs out of guest passes and tries to admit another guest. * Test data:   “Member First Name: Jonnathan  Member Last Name: Wei  Member Date of Birth: 9/21/1992  Class Location: bridgewater  Class Name: cardio  Instructor Name: Davis  Member or Guest? Guest”  Click Check In four times | “Jonnathan Wei ran out of guest pass.” |