Test Design Document

Authors: Anirudh Chauhan, Matthew Calora

Date class, isValid() method

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case # | Requirement | Test description and Input data | Expected result/output |
| 1 | Must take a date of birth that is not later than the current day and the date must be a valid date | Boolean beforeCurrentDay must be false from afterCurrentDay()  Boolean validateDay must return true from validatedate()  Returns true if the date is valid  Test data 1: 12/3/2025  Test data 2: 12/3/1990 | 1: false  2: true |
| 2 | Number of days in February for a non-leap year shall be 28 | Takes month and day and uses it in validateDate(). Month must be 2. This checks if the day is a valid February date. If not, output false  Test data 1: 2/29/2018  Test data 2: 2/15/2018 | 1: False  2: True |
| 3 | Valid range for the month shall be 1-12 | Takes month and day and uses it in validateDate(). If month is not within the range of 1-12, return false  Test data 1: 13/17/2021  Test data 2: 1/17/2021 | 1: False  2: True |
| 4 | Valid range for the day shall be 1-31 | Day in validateDate(). If day is not within the range of 1-31, return false  Test data1: 1/45/2020  Test data 2: 1/4/2020 | 1: false  2: true |

Member class, compareTo() method

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case # | Requirement | Test description and Input data | Expected result/output |
| 1 | From member input, member must first exist in the database | If member is not in database, return 0  If member is in database, return 1  Test data1: John Doe (not in database)  Test data2: John Doe (already in database) | Test1: 0  Test2: 1 |
| 2 |  |  |  |
| 3 |  |  |  |

Validation() method

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case # | Requirement | Test description and input data | Expected result |
| 1 | Date of birth must be valid (date exists and date is not the current date) | If date exists and is not current date, return true. Else return false  Test1: 01/01/2064  Test2: 09/15/2001 | T1: false  T2: true |
| 2 | Expiration date for membership must still be valid at the time of enrollment | If expiration date is not expired, return true. Else, return false  Test1: 11/7/2022 (expires today)  Test2: 11/15/2025  (not expired) | T1: false  T2: true |
| 3 | Member must be above the age of 18 | If member is above the age of 18, return true. Else return false  Test1: member is 9 years old  Test2: member is 21 years old | T1: false  T2: true |
| 4 | Location must be one of the locations listed in the database | If location input is listed in the database, return true. Else return false  Test1: Ridgewood  Test2: Piscataway | T1: false  T2: true |

validCheckIn() method

|  |  |  |  |
| --- | --- | --- | --- |
| Test # | Requirement | Test description and Input data | Expected result/output |
| 1 | Expiration date for membership must still be valid at the time of enrollment | If expiration date is not expired, return true. Else, return false  Test1: 11/7/2022 (expires today)  Test2: 11/15/2025  (not expired) | T1: false  T2: true |
| 2 | Member must not have a time conflict with another fitness class | If the member does not have a time conflict without another fitness class, return false. Else return true  Test1: Member has time conflict  Test2: Member does not have time conflict | T1: false  T2: true |
| 3 | Member’s membership rank determines if they can enroll in fitness classes that are not within their main gym location | If membership is Standard or Family, return false. If membership is premium, return true  Test1: Member has standard membership  Test2: Member has family membership  Test3: Member has premium membership | T1: false  T2: false  T3: true |

validGuestCheckIn() method

|  |  |  |  |
| --- | --- | --- | --- |
| Test # | Requirement | Test description and Input data | Expected result/output |
| 1 | Expiration date for membership must still be valid at the time of enrollment | If expiration date is not expired, return true. Else, return false  Test1: 11/7/2022 (expires today)  Test2: 11/15/2025  (not expired) | T1: false  T2: true |
| 2 | Guest must not have a time conflict with another fitness class | If the member does not have a time conflict without another fitness class, return false. Else return true  Test1: Member has time conflict  Test2: Member does not have time conflict | T1: false  T2: true |
| 3 | Member’s membership rank determines if they can enroll in fitness classes that are not within their main gym location. This restriction also applies to guests whose member’s rank is not high enough. | If membership is Standard or Family, return false. If membership is premium, return true  Test1: Guest’s member has standard membership  Test2: Guest’s member has family membership  Test3: Guest’s member has premium membership | T1: false  T2: false  T3: true |

validConditions() method (for fitness classes)

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case # | Requirement | Test description and input data | Expected result |
| 1 | Date of birth must be valid (date exists and date is not the current date) | If date exists and is not current date, return true. Else return false  Test1: 01/01/2064  Test2: 09/15/2001 | T1: false  T2: true |
| 2 | Expiration date for membership must still be valid at the time of enrollment | If expiration date is not expired, return true. Else, return false  Test1: 11/7/2022 (expires today)  Test2: 11/15/2025  (not expired) | T1: false  T2: true |
| 3 | Member must be above the age of 18 | If member is above the age of 18, return true. Else return false  Test1: member is 9 years old  Test2: member is 21 years old | T1: false  T2: true |
| 4 | Location must be one of the locations listed in the database | If location input is listed in the database, return true. Else return false  Test1: Ridgewood  Test2: Piscataway | T1: false  T2: true |
| 5 | Member must be in the database | If member is in the database, return true. Else return false  T1: not in database  T2: in database | T1: false  T2: true |
| 6 | Fitness class given must exists and must be selected in the GUI | If fitness class given exists in database, return true. Else return false  T1: input not selected  T2: any input is selected | T1: false  T2: true |
| 7 | Fitness class instructor must exist | If fitness class instructor exists, return true. Else return false  Test1: instructor does not exist  Test2: instructor exists | T1: false  T2: true |
| 8 | The fitness class must exist at a certain location. Some fitness classes are not given at some locations | If the fitness class selected is being given at a certain location, return true. Else return false  Test1: Pilates at Somerville  Test2: Pilates at Piscataway | T1: false  T2: true |