

Lab 3

Andrew Hopkins

CM: 1164

1. The source code is found in the Expedia project, and test code is found in the ExpediaTest project.
2. The Flight class allows users to “book” flight by creating a new Flight object. It also adds functionality to see if two flights are the same.
3. The classes are:
 - AssemblyInfo
 - Booking
 - Car
 - Flight
 - Hotel
 - User
 - Booking
4. The test classes are:
 - BookingTest
 - CarTest
 - FlightTest
 - HotelTest
 - UserTest
5. The test methods in UserTest class are:
 - TestThatUserInitializes
 - TestThatUserHasZeroFrequentFlierMilesOnInit
 - TestThatUserCanBookEverything
 - TestThatUserHasFrequentFlierMilesAfterBooking
 - TestThatUserCanBookAFlight
 - TestThatUserCanBookAHotelAndACar
 - TestThatUserHasCorrectNumberOfFrequentFlyerMilesAfterOneFlight
6. Some of the possible assert methods are: AreEqual, DoesNotThrow, and IsTrue.
7. AreEqual checks to make sure the two parameters have the same value. DoesNotThrow checks to make sure method does not throw an exception. IsTrue checks to make sure the parameter is true.
8. AreEqual checks to make sure the two parameters have the same value while AreSame checks to make sure the two parameters point to the same object.
9. TestThatHotelInitializes tests to make sure the constructor returns something, it doesn't matter what it is, as long as it is not null.
10. The algorithm simply returns a value equal to 45 * the number of nights the user is renting.
11. The new tests test to make sure the getBasePrice function works for varying number of days.
12. You don't have to check if the values are not null because the other tests make sure the value is equal to some number, so clearly 450 is not equal to null.
13. This test expects an exception because in the Hotel constructor if the number of nights to stay is less than 1 the constructor throws an exception.
14. `[ExpectedException(typeof(OutOfMemoryException))]`