Kevin Risden CM1328 3/15/11
CSSE376

Lab 3 – Answers

1. The source code is found in the Expedia project and the test code is found in the ExpediaTest project.
2. In the Expedia project there are 5 classes: Booking, Car, Flight, Hotel, and User.
3. The Flight class supports initializing a Flight with a startDate and endDate along with miles. It will calculate the booking price and when asked it can return the Miles for a specific flight. The Flight class will also do error checking to make sure end is after start date and miles is not negative.
4. In the ExpediaTest project there are also 5 classes: BookingTest, CarTest, FlightTest, HotelTest, and UserTest.
5. In the UserTest class the test methods are TestThatUserInitializes(), TestThatUserHasZeroFrequentFlierMilesOnInit(),TestThatUserCanBookEverything(),TestThatUserHasFrequentFilerMilesAfterBooking(), TestThatUserCanBookAFlight(), TestThatUserCanBookAHotelAndACar(), TestThatUserHasCorrectNumberOfFrequentFlyerMilesAfterOneFlight().
6. Three functions supported by the Assert class are AreEqual(), AreNotEqual(), and AreSame().
7. AreEqual() checks that two objects are equal in value. AreNotEqual() checks that two objects are not equal in value. AreSame() checks that two objects reference the same memory location and are the same object.
8. AreEqual check that the value of the objects are the same so that they are two separate instances of the same object. Where as, AreSame checks that it is the same memory reference for both objects and therefore are the same instance of an object.
9. The unit test TestThatHotelInitializes makes sure that the constructor for Hotel works and actually initializes the hotel object.
10. The generic algorithm for calculating getBasePrice is 45*numberOfNightsToRent.
11. The cases tested with the new test cases are that the correct base price is calculated for 1 day, 2 days, and 10 days.
12. The tests are ran in order and since the TestThatHotelInitializes() is run before the base price tests then there is no need to check it again.
13. This should throw an error due to the constructor for Hotel checking that nightsToRent > 0 and if not it throws the error ArgumentOutOfRangeException.
14. [ExpectedException(typeof(OutOfMemoryException))]