1. The source code is found in "Expedia" The test code is in "ExpediaTest"
2. The classes that exist in Expedia is: AssemblyInfo, Booking, Car, Flight, Hotel, User.
3. The flight class is constructed with a startdate and an enddate. It can check if two flights are equal if they arrive at the same time, leave at the same time and travel the same distance.
4. BookingTest, CarTest, FlgihtTest, HotelTest, UserTest
5. TestThatUSerInitializes, TestThatUserHasZeroFrequenceFlierMilesOnInit, TestThatUSerCanBookEverything, TestThatUserHasFrequentFlierMilesAfterBooking, TestThatUserCanBookAFlight, TestThatUserCanBookAHotelAndACar, TestThatUserHasCorrectNumberOfFrequentFlyerMilesAfterOneFlight.
6. AreEqual, AreNotEqual, False
7. AreEqual takes two values and checks to see if they are the same. AreNotEqual does the same but checking if they are different. False checks to make sure the Boolean given to it is false.
8. AreEqual checks their values to see if they are the same and AreSame checks to see if they are the exact same object.
9. The test is verifying that a Hotel Object is actually made.
10. The algorithm for get base price 45* the number of nights staying.
11. The new tests are verifying that the getBasePrice() function is working properly. This is done by testing: 1, 2 and 10.
12. We have verified that hotels can be created by the first test. If it fails the first test then the initialization doesn't work. Since that has been verified we can test other functions without worrying.
13. Because you can't stay at a hotel for a negative number of nights. That's why it is throwing an error.
14. [Test()]
    [ExpectedException(typeof(OutOfMemoryException)]