1. Expedia is the source code, and ExpediaTest is the test code
2. The classes are Car, Flight, Hotel, and User.
3. It allows one to specify a flight, with start date, end date, distance to be traveled, and a method to compare it to other flights/objects.
4. There are test classes: BookingTest, CarTest, FlightTest, HotelTest, and UserTest.
5. The test methods included are: TestThatUserInitializes, TestThatUserHasZeroFrequentFlierMilesOnInit, TestThatUserCanBookEverything, TestThatUserHasFrequentFlierMilesAfterBooking, TestThatUserCanBookAFlight, TestThatUserCanBookAHotelAndACar, TestThatUserHasCorrectNumberOfFrequentFlyerMilesAfterONeFlight.
6. Three Assert methods are: AreEqual, AreNotEqual, and AreNotSame.
7. AreEqual, tests to see if the attributes of two different objects are the same, AreNotEqual tests the same but returns the oppposite in the same conditions, and AreNotSame makes sure that the two objects do not have the same memory reference.
8. AreEqual checks to see if the attributes of two objects are the same, if the two objects are equal, interchangeable. AreSame checks if they share the same memory reference, that they are the same actual object.
9. The method tests that the Hotel class can initialize a new Hotel Object.
10. 45 multiplied by the number of nights to rent.
11. That the base price calculations are correct for 1, 2, and 10 days.
12. We create a new target in each test, and the initialization method was tested in the first test, so there is no need to test that again.
13. Since the Hotel class throws an exception if the nights to be rented is negative.
14. [ExpectedException(typeof(OutOfMemoryException))]