Robin Cooper

Lab 3 – Unit testing

Question 1: The code is found in the Expedia project. The test code is found in ExpediaTest.

Question 2: AssemblyInfo, Booking, Car, Flight, Hotel, User. Tests exist as BookingTest, CarTest, FlightTest, HotelTest, UserTest.

Question 3: The Flight class constructs and validates a Flight to be booked. It stores start end and miles.

Question 4: BookingTest, CarTest, FlightTest, HotelTest, UserTest.

Question 5: There's a setup, a teardown, and a bunch of tests (you don't really want me to list them all do you?).

Question 6: AreEqual AreSame Throws. Respectively checks if things are equal, checks if pointers are to the same object, and checks if the code throws an error.

Question 7: AreEqual AreSame Throws. Respectively checks if things are equal, checks if pointers are to the same object, and checks if the code throws an error.

Question 8: AreEqual AreSame. Respectively checks if things are equal and checks if pointers are to the same object.

Question 9: It verifies that creation of the Hotel object was successful.

Question 10: Base price is 45 per night.

Question 11: The new test check the base price over the first three days of stay to check if it's calculated properly.

Question 12: We don't need to include Assert.NotNull because the `TestThatHotelInitializes` function already took care of that test.

Question 13: The test expects an exception because staying for a negative amount of days is illegal.

Question 14: `[ExpectedException(typeof(OutOfMemoryException))]`