Obstacles overcome:

* I was initially confused about how to count miles beyond 100 and up to 500 before I realized it was pretty simple and you just had to count the miles between 100 and 500. This is the same logic for the miles between 500 and 600
* I was a little confused about how cin.ignore worked but after using it I finally understood why it was necessary when switching from cin to getline
* I was briefly confused as to why my program was skipping two lines between each statement (each “cout”) before realizing that the user was entering information which was automatically a new line

Test Cases:

* Simple test case from Professor Smallberg to test that program works and can calculate correctly: (2417, 2754, 4, Mario Andretti, n, 10) → (“The rental charge for Mario Andretti is $276.99”)
* Another simple test case from Professor Smallberg to test that the program works and can calculate correctly: (1885, 1973, 1, Danica Patrick, y, 2) → “The rental charge for Danica Patrick is $104.04”
* Test case to test error for negative starting odometer value: (-1) → (“The starting odometer reading must not be negative.”)
* Test case to test error for final odometer value being lesser than initial odometer value: (10, 5) → (“The ending odometer reading must be at least as large as the starting reading.”)The ending odometer reading must be at least as large as the starting reading.
* Test case to test error for number of days car was rented being negative: (10, 15, -2) → (“The number of rental days must be positive.”)
* Test case to test error for no text input for customer name: (1,2,3,“”)-->(“You must enter a customer name”)
* Test case to test error if luxury status response is not “y” or “n”: (1,2,3,“Mary”, “pop”)-->(“You must enter y or n.”)You must enter y or n.
* Test case to test error if month number is not an integer between 1 and 12 inclusive: (1,2,3,”Mary”, “y”, “13”) → (“The month number must be in the range 1 through 12.”)
* Test case to test program logic for miles over 600: (1000, 2000, 5, Mario, y, 1) → (“The rental charge for Mario is $657”)
* Test case to test program logic for miles between 100 and 600 (1000, 1500, 5, Mario, y, 1) → (“The rental charge for Mario is $540”)