

# Vehicle Rental Manager Black Box Test Plan

---

**Document Author(s):** Noah Benveniste

**Date:** 4 April 2018

---

## Introduction

This test suite focuses on testing the basic, overarching functionality of the Vehicle Rental Manager program. To start the program, the user runs VehicleRentalManagerUI.java. The tests that use a valid input file will use input/sample.csv, the contents of which are available below. The tests focus on valid and invalid startup of the program, invalid and valid command selection, and output for generating the optimal sequence of rentals and querying the rentals for a given day.

input/sample.csv

START\_DAY,END\_DAY,COST,MAKE,MODEL

1,2,85,[Chevrolet](#),[Tahoe](#)

1,4,255,[Toyota](#),[Prius](#)

2,5,220,Ford,Explorer

4,5,50,Honda,Accord

2,3,65,Jeep,Compass

3,5,90,Ford,Explorer

3,4,55,[Kia](#),Soul

1,5,500,Honda,CRV

1,3,180,[Chevrolet](#),[Silverado](#)

2,4,90,Jeep,[Cherokee](#)

Test ID	Description	Expected Results	Actual Results
<b>testStartInvalidInput (DT)</b>	Preconditions: <i>None</i>  The user runs VehicleRentalManagerUI.java  When prompted for the input file, the user enters the following:  "vehicleDNE.csv"  Check results	The user is initially prompted with the following:  "Enter path to rental cost file: "  After entering the invalid file path, the following error message is displayed:  <u>java.io.FileNotFoundException:</u> vehicleDNE.csv (The system cannot find the file specified)	The user is initially prompted with the following:  "Enter path to rental cost file: "  After entering the invalid file path, the following error message is displayed:  <u>java.io.FileNotFoundException:</u> vehicleDNE.csv (The system cannot find the file specified)

		The user is then reprompted to enter the input file.	The user is then reprompted to enter the input file.
<b>testStartValidInput (ECP)</b>	<p>Preconditions: <i>None</i></p> <p>The user runs VehicleRentalManagerUI.java</p> <p>When prompted for the input file, the user enters the following:</p> <p>"input/sample.csv"</p> <p>Check results</p>	<p>The user is initially prompted with the following:</p> <p>"Enter path to rental cost file: "</p> <p>After entering the file path, the user is given the following prompt:</p> <p>"Enter 1 to get the cost-minimized sequence of car rentals between two days or 2 to list all of the available rentals for a given day: "</p>	<p>The user is initially prompted with the following:</p> <p>"Enter path to rental cost file: "</p> <p>After entering the file path, the user is given the following prompt:</p> <p>"Enter 1 to get the cost-minimized sequence of car rentals between two days or 2 to list all of the available rentals for a given day: "</p>
<b>testInvalidInputGUI (DT)</b>	<p>Preconditions: <i>testStartValidInput passes</i></p> <p>(Run testStartValidInput)</p> <p>The user is prompted with the following:</p> <p>"Enter 1 to get the cost-minimized sequence of car rentals between two days or 2 to list all of the available rentals for a given day: "</p> <p>Enter 3</p>	<p>The following is output:</p> <p>"Invalid input.</p> <p>Run again? (y/n):"</p>	<p>The following is output:</p> <p>"Invalid input.</p> <p>Run again? (y/n):"</p>

	Check results		
<b>testOptimalRentalSequence AvailableRentals (ECP)</b>	<p>Preconditions: <i>testStartValidInput passes</i></p> <p>(Run testStartValidInput)</p> <p>The user is prompted with the following:</p> <p>“Enter 1 to get the cost-minimized sequence of car rentals between two days or 2 to list all of the available rentals for a given day: “</p> <p>Enter <b>1</b></p> <p>The user is prompted to enter the start day. Enter <b>1</b></p> <p>The user is prompted to enter the end day. Enter <b>5</b></p> <p>Check results</p>	<p>The program outputs the following:</p> <p>Rental total is \$225.00</p> <p>[     From day 1 to day 2: \$85.00, Chevrolet Tahoe     From day 2 to day 4: \$90.00, Jeep Cherokee     From day 4 to day 5: \$50.00, Honda Accord ]</p>	<p>The program outputs the following:</p> <p>Rental total is \$225.00</p> <p>[     From day 1 to day 2: \$85.00, Chevrolet Tahoe     From day 2 to day 4: \$90.00, Jeep Cherokee     From day 4 to day 5: \$50.00, Honda Accord ]</p>
<b>testOptimalRentalSequence UnavailableRentals (BVA)</b>	<p>Preconditions: <i>testStartValidInput passes</i></p> <p>(Run testStartValidInput)</p>	<p>The program outputs the following:</p> <p>Rental total is \$140.00</p> <p>[</p>	<p>The program outputs the following:</p> <p>Rental total is \$140.00</p> <p>[</p>

	<p>The user is prompted with the following:</p> <p>“Enter 1 to get the cost-minimized sequence of car rentals between two days or 2 to list all of the available rentals for a given day: “</p> <p>Enter <b>1</b></p> <p>The user is prompted to enter the start day. Enter <b>2</b></p> <p>The user is prompted to enter the end day. Enter <b>7</b></p> <p>Check results</p>	<p>From day 2 to day 4: \$90.00, Jeep Cherokee</p> <p>From day 4 to day 5: \$50.00, Honda Accord</p> <p>No rentals available on day 5 ]</p>	<p>From day 2 to day 4: \$90.00, Jeep Cherokee</p> <p>From day 4 to day 5: \$50.00, Honda Accord</p> <p>No rentals available on day 5 ]</p>
<p><b>testQuery</b> <b>(ECP)</b></p>	<p>Preconditions: <i>testStartValidInput passes</i></p> <p>(Run testStartValidInput)</p> <p>“Enter 1 to get the cost-minimized sequence of car rentals between two days or 2 to list all of the available rentals for a given day: “</p>	<p>The program outputs the following:</p> <p>Available rentals for day 1 \$85.00 Chevrolet Tahoe for day 1 to day 2 \$180.00 Chevrolet Silverado for day 1 to day 3 \$255.00 Toyota Prius for day 1 to day 4 \$500.00 Honda CRV for day 1 to day 5</p>	<p>The program outputs the following:</p> <p>Available rentals for day 1 \$85.00 Chevrolet Tahoe for day 1 to day 2 \$180.00 Chevrolet Silverado for day 1 to day 3 \$255.00 Toyota Prius for day 1 to day 4 \$500.00 Honda CRV for day 1 to day 5</p>

	<p>Enter 2</p> <p>The user is prompted to enter the day to query for. Enter 1</p> <p>Check results</p>	]	]
<b>testQuit (ECP)</b>	<p>Preconditions: <i>testStartValidInput</i>, <i>testInvalidInputGUI</i> both pass</p> <p>(Run <i>testStartValidInput</i> and <i>testInvalidInputGUI</i>)</p> <p>The following is output:</p> <p>“Invalid input. Run again? (y/n):”</p> <p>Type n</p> <p>Check results</p>	The program exits successfully	The program exits successfully

#### Document Revision History

Date	Author	Change Description
4/4/18	Noah Benveniste	<ul style="list-style-type: none"> <li>Ran black box tests</li> </ul>

