

For the integration verification tests, we use a partition of inputs to test relevant inputs and determine if the actual outputs correspond to our expected outputs. For the first integration test, we have a user enter in the criteria for a previous rental contract, and then when the contract has been successfully found, the user receives an email with the requested information. In our first test case for that scenario, the user enters the correct information and receives the emailed receipt; in the second test case the user enters invalid information and is not able to receive the receipt but instead gets a message indicating the receipt was not found and to try again. In our second integration test, we have an employee user schedule maintenance for a vehicle currently in the inventory. In test case 1 the user enters the information for a vehicle that is currently in inventory, and successfully schedules the maintenance; in the second test case the user schedules maintenance for a vehicle that is not currently in inventory and receives a message indicating the vehicle is currently in use. All tests conducted passed and did not have any errors that required adjusting or fixing.

Integration 1 Test	Integration 2 Test
<p>Feature: ClientHistory viewPreviousRentals(RentalContract)</p> <p>Test case 1: Scenario: A client wants to receive the receipt from a previous rental agreement and enters the correct criteria Input:</p> <ul style="list-style-type: none"> - agreementID: 123ABC - ClientID: BillyBob43 - vehicleID: 5DB89 - startDate: 10-31-2024 - endDate: 11-15-2024 <p>ClientHistory viewPreviousRentals(RentalContract): void</p> <p>RentalContract rc = new RentalContract; rc.agreementID = "123ABC"; rc.clientID = "BillyBob43"; rc.vehicleID = "5DB89"; rc.startDate = 10312023; rc.endDate = 11152023;</p>	<p>Feature: Inventory scheduleMaintenance()</p> <p>Test case 1: Scenario: Employee schedules maintenance for a vehicle that is currently available in inventory. Input:</p> <ul style="list-style-type: none"> - vehicleID: 2AV55 - make: Ford - model: Explorer <p>Inventory scheduleMaintenance(): void</p> <p>Vehicle v = new Vehicle; v.vehicleID = "2AV55"; v.make = "Ford"; v.model = "Explorer";</p> <p>if (v.checkAvailability() == true){ v.scheduleMaintenance(); print: v + "has been scheduled for maintenance"; } else { print: v + "is currently in use and cannot be scheduled for maintenance";</p>

<pre> if (viewPreviousRentals(RentalContract) == rc){ rc.sendReceipt(); print: rc + "receipt has been sent"; } else { print: "Could not locate previous rental agreement, please try again"; } </pre> <p>Output:</p> <ul style="list-style-type: none"> - "123ABC BillyBob43 5DB89 10312023 11152023 receipt has been sent." <p>Test result: Pass</p> <p>-----</p> <p>-</p> <p>Test case 2: Scenario: A client wants to receive the receipt from a previous rental agreement and enters incorrect criteria Input:</p> <ul style="list-style-type: none"> - agreementID: 321CBA - ClientID: Bob - vehicleID: 5DB89 - startDate: 10-31-2024 - endDate: 11-15-2024 <p>ClientHistory viewPreviousRentals(RentalContract): void</p> <p>RentalContract rc = new RentalContract; rc.agreementID = "123ABC"; rc.clientID = "BillyBob43"; rc.vehicleID = "5DB89"; rc.startDate = 10312023; rc.endDate = 11152023;</p> <pre> if (viewPreviousRentals(RentalContract) == rc){ rc.sendReceipt(); print: rc + "receipt has been sent"; } else { </pre>	<pre> } </pre> <p>Output:</p> <ul style="list-style-type: none"> - checkAvailability() = true; - "2AV55 Ford Explorer has been scheduled for maintenance." <p>Test result: Pass</p> <p>-----</p> <p>-</p> <p>Test case 2: Scenario: Employee schedules maintenance for a vehicle that is not currently available in the inventory. Input: 3EF35, Dodge, Viper</p> <p>Inventory scheduleMaintenance(): void</p> <p>Vehicle v = new Vehicle; v.vehicleID = "3EF35"; v.make = "Dodge"; v.model = "Viper";</p> <pre> if (v.checkAvailability() == true){ v.scheduleMaintenance(); print: v + "has been scheduled for maintenance"; } else { print: v + "is currently in use and cannot be scheduled for maintenance"; } </pre> <p>Output:</p> <ul style="list-style-type: none"> - checkAvailability() = false; - "3EF35 Dodge Viper is currently in use and cannot be scheduled for maintenance." <p>Test result: Pass</p>
---	---

<pre>print: "Could not locate previous rental agreement, please try again"; }</pre> <p>Output:</p> <ul style="list-style-type: none">- "Could not locate previous rental agreement, please try again." <p>Test result: Pass</p>	
---	--