

Date	Test Description	Files	Inputs	Expected output	Actual output	Notes
23/4/19	Add a new room type	room.rb	single = Room.new("single", "feature here", 200, "everyday") p single	#<Room:0x00007fc94d907ea0 @type="single", @features="feature here", @price=200, @availability="everyday">	as expected	
23/4/19	Add a new Deluxe room	room.rb	deluxe = Deluxe.new p deluxe	#<Deluxe:0x00007fc94d907c98 @type="Deluxe", @features="* Extra comfy bedding \n* One Story Scratching Post \n* Lots of nummy treats", @price=300, @availability={:Monday=>"Available", :Tuesday=>"Available", :Wednesday=>"Available", :Thursday=>"Available", :Friday=>"Available", :Saturday=>"Available", :Sunday=>"Available"}>	as expected	
23/4/19	Add a new Luxury room	room.rb	luxury = Luxury.new p luxury	#<Luxury:0x00007f88509f3848 @type="Luxury", @features="features here", @price=350, @availability={:Monday=>"Available", :Tuesday=>"Available", :Wednesday=>"Available", :Thursday=>"Available", :Friday=>"Available", :Saturday=>"Booked Out", :Sunday=>"Booked Out"}>	as expected	
23/4/19	Add a new hotel	hotel.rb	hotel = Hotel.new("Purr Seasons", "Address here", "1800 MEOOW MEOOW") p hotel	#<Hotel:0x00007fd62d134998 @name="Purr Seasons", @address="Address here", @phone="1800 MEOOW MEOOW", @rooms=[]>	as expected	
23/4/19	Add a new cat guest	cat.rb	cat = Cat.new("Smokey", 4) p cat	#<Cat:0x00007fedd38ccd78 @name="Smokey", @age=4, @booking=false>	as expected	
23/4/19	Add a new booking	booking.rb	booking = Booking.new(Luxury.new) p booking	#<Booking:0x00007fd58282a9b0 @room=#<Luxury:0x00007fd58282ab68 @type="Luxury", @features="features here", @price=350, @availability={:Monday=>"Available", :Tuesday=>"Available", :Wednesday=>"Available", :Thursday=>"Available", :Friday=>"Available", :Saturday=>"Booked Out", :Sunday=>"Booked Out"}>, @days=[]>	as expected	
23/4/19	Add a new Deluxe and Luxury room to the hotel	hotel.rb	hotel = Hotel.new hotel.add_room(Deluxe.new). add_room(Luxury.new) p hotel	#<Hotel:0x00007fe05f922920 @name="Purr Seasons", @address="32 Cat St", @phone="1800 MEOOW MEOOW", @rooms=[#<Deluxe:0x00007fe05f922858 @type="Deluxe", @features="features here", @price=300, @availability={:Monday=>"Available", :Tuesday=>"Available", :Wednesday=>"Available", :Thursday=>"Available", :Friday=>"Available", :Saturday=>"Available", :Sunday=>"Available"}>, #<Luxury:0x00007fe05f9226a0 @type="Luxury", @features="features here", @price=350, @availability={:Monday=>"Available", :Tuesday=>"Available", :Wednesday=>"Available", :Thursday=>"Available", :Friday=>"Available", :Saturday=>"Booked Out", :Sunday=>"Booked Out"}>]>	as expected	
23/4/19	Display list of rooms	hotel.rb	puts hotel.display_rooms	* Deluxe * Luxury	* Deluxe * Luxury #<Deluxe:0x00007f8781889ba8> #<Luxury:0x00007f8781889950>	Both displaying and printing room types. Puts already in method need to run test without puts.
23/4/19	Display list of rooms	hotel.rb	hotel.display_rooms	* Deluxe * Luxury	as expected	
23/4/19	Add a new hotel with rooms	main.rb; room.rb; hotel.rb	hotel = Hotel.new.add_room(Deluxe.new).add_room(Luxury.new) p hotel	#<Hotel:0x00007fc9f7a11e40 @name="Purr Seasons", @address="32 Cat St", @phone="1800 MEOOW MEOOW", @rooms=[#<Deluxe:0x00007fc9f7a11d78 @type="Deluxe", @features="features here", @price="300.00", @availability={:Monday=>"Available", :Tuesday=>"Available", :Wednesday=>"Available", :Thursday=>"Available", :Friday=>"Available", :Saturday=>"Available", :Sunday=>"Available"}>, #<Luxury:0x00007fc9f7a11af8 @type="Luxury", @features="features here", @price="350.00", @availability={:Monday=>"Available", :Tuesday=>"Available", :Wednesday=>"Available", :Thursday=>"Available", :Friday=>"Available", :Saturday=>"Booked Out", :Sunday=>"Booked Out"}>]>	as expected	
23/4/19	Add a new cat	main.rb; cat.rb	cat = Cat.new(cat_name) p cat	#<Cat:0x00007fc9f7a11558 @name="Sly", @booking=nil>	as expected	
23/4/19	Display days selected as an array.	room.rb	luxury = Luxury.new p luxury.select_days	["Tuesday", "Wednesday", "Thursday"]	as expected	

Date	Test Description	Files	Inputs	Expected output	Actual output	Notes
23/4/19	Testing display_booking method	booking.rb	booking = Booking.new("donut", ["monday", "tuesday", "wednesday"]) booking.display_booking		1: from booking.rb:22:in `<main>' booking.rb:12:in `display_booking': undefined method `type' for "donut": String (NoMethodError)	
23/4/19	Testing display_booking method	booking.rb	booking = Booking.new(Luxury.new, ["monday", "tuesday", "wednesday"]) booking.display_booking	Your booking: Room Type: Luxury Booking Days: monday tuesday wednesday	as expected	
24/4/19	Test when the user enter no name for the cat	main.rb; cat.rb	At prompt: What is your cat's name please? Press 'enter' without typing a name for their cat	Ooops, please enter your cat's name:	as expected - user is repeatedly asked to enter something if user doesn't enter anything. But we want a way for user to exit the endless loop	Give the user an option to move past this without entering a cat name.
24/4/19	Test when the user selects days out of order	main.rb; room.rb	At prompt: Please select the days you would like to stay: Select days in the wrong order by arrowing up.	Booking Days: * Tuesday * Wednesday * Thursday	Booking Days: * Thursday * Wednesday * Tuesday	BUG added to Trello: Need to consider how to display days in order.
24/4/19	Test when the user selects no days for the booking	main.rb; room.rb	At prompt: Please select the days you would like to stay: Enter without selecting any days.	Error	Booking Days: ----- ----- 0 days @ \$270.00 each	BUG added to Trello: Need to consider what happens if the user enters no days - it shouldn't create a booking with 0 days.
25/4/19	Test when a user makes a booking that availability for that room changes to reflect days now booked out	main.rb; room.rb	Made a booking in a Deluxe room for Tuesday, Wednesday, Thursday, Friday.	DELUXE ROOM Availability: * Monday: Available * Tuesday: Booked Out * Wednesday: Booked Out * Thursday: Booked Out * Friday: Booked Out * Saturday: Available * Sunday: Available	as expected	