Michael Newman

Sunday, October 6, 2019

Assignment 2

Differences from assignment 2

# Controller Updates

## Car controller update to enable search

diff --git a/app/controllers/cars\_controller.rb b/app/controllers/cars\_controller.rb

index a1b7200..45c8e69 100644

--- a/app/controllers/cars\_controller.rb

+++ b/app/controllers/cars\_controller.rb

@@ -1,7 +1,7 @@

class CarsController < ApplicationController

before\_action :set\_car, only: [:show, :edit, :update, :destroy]

- autocomplete :car, :model, full\_search: true

+ # autocomplete :car, :vin, full\_search: true

# GET /cars

# GET /cars.json

@@ -31,8 +31,6 @@ class CarsController < ApplicationController

# POST /cars.json

def create

@car = Car.new(car\_params)

- @makes = Make.all

- @parts = Part.all

respond\_to do |format|

if @car.save

@@ -71,7 +69,7 @@ class CarsController < ApplicationController

# GET /cars/search

def search

- @cars = Car.where("model like ?", "%#{params[:search]}%")

+ @cars = Car.where("vin like ?", params[:search])

render :index

end

## Make controller update to enable search

diff --git a/app/controllers/makes\_controller.rb b/app/controllers/makes\_controller.rb

index 61b146e..19460d9 100644

--- a/app/controllers/makes\_controller.rb

+++ b/app/controllers/makes\_controller.rb

@@ -1,8 +1,6 @@

class MakesController < ApplicationController

before\_action :set\_make, only: [:show, :edit, :update, :destroy]

- autocomplete :make, :name, full\_search: true

-

# GET /makes

# GET /makes.json

def index

@@ -63,11 +61,6 @@ class MakesController < ApplicationController

end

end

- def search

- @makes = Make.where("name like ?", "%#{params[:search]}%")

- render :index

- end

-

private

# Use callbacks to share common setup or constraints between actions.

def set\_make

## Parts controller update to enable search

diff --git a/app/controllers/parts\_controller.rb b/app/controllers/parts\_controller.rb

index 400a2c1..2e52866 100644

--- a/app/controllers/parts\_controller.rb

+++ b/app/controllers/parts\_controller.rb

@@ -1,8 +1,6 @@

class PartsController < ApplicationController

before\_action :set\_part, only: [:show, :edit, :update, :destroy]

- autocomplete :part, :name, full\_search: true

-

# GET /parts

# GET /parts.json

def index

@@ -63,11 +61,6 @@ class PartsController < ApplicationController

end

end

- def search

- @parts = Part.where("name like ?", "%#{params[:search]}%")

- render :index

- end

-

private

# Use callbacks to share common setup or constraints between actions.

def set\_part

# Validations

## Car validations

diff --git a/app/models/car.rb b/app/models/car.rb

index 9317d93..b636ddc 100644

--- a/app/models/car.rb

+++ b/app/models/car.rb

@@ -1,8 +1,4 @@

class Car < ApplicationRecord

belongs\_to :make

has\_and\_belongs\_to\_many :parts

-

- validates :vin, presence: true, numericality: true, uniqueness: true

- validates :model, presence: true

- validates :make\_id, presence: true

end

## Make Validations

diff --git a/app/models/make.rb b/app/models/make.rb

index 2273438..c53d11d 100644

--- a/app/models/make.rb

+++ b/app/models/make.rb

@@ -1,6 +1,2 @@

class Make < ApplicationRecord

- has\_many :cars

-

- validates :name, presence: true

- validates :country, presence: true

End

## Part Validations

diff --git a/app/models/part.rb b/app/models/part.rb

index 2455f40..72dc3f4 100644

--- a/app/models/part.rb

+++ b/app/models/part.rb

@@ -1,5 +1,3 @@

class Part < ApplicationRecord

has\_and\_belongs\_to\_many :cars

-

- validates :name, presence: true

end

# Index updates to add search bar

## Car index

diff --git a/app/views/cars/index.html.erb b/app/views/cars/index.html.erb

index 125b646..a4c9c4a 100644

--- a/app/views/cars/index.html.erb

+++ b/app/views/cars/index.html.erb

@@ -31,11 +31,3 @@

<br>

<%= link\_to 'New Car', new\_car\_path %>

-<br>

-

-<%= form\_tag(search\_cars\_url, method: "get") do %>

- <%= label\_tag(:search, "Search by Model:") %>

- <%= text\_field\_tag :search, params[:search],

- data: { autocomplete: autocomplete\_car\_model\_cars\_path } %>

- <%= submit\_tag("Search") %>

-<% end %>

## Make Index

diff --git a/app/views/makes/index.html.erb b/app/views/makes/index.html.erb

index 77fc94d..f675780 100644

--- a/app/views/makes/index.html.erb

+++ b/app/views/makes/index.html.erb

@@ -27,10 +27,3 @@

<br>

<%= link\_to 'New Make', new\_make\_path %>

-<br>

-<%= form\_tag(search\_makes\_url, method: "get") do %>

- <%= label\_tag(:search, "Search by Make Name:") %>

- <%= text\_field\_tag :search, params[:search],

- data: { autocomplete: autocomplete\_make\_name\_makes\_path } %>

- <%= submit\_tag("Search") %>

-<% end %>

## Part Index

diff --git a/app/views/parts/index.html.erb b/app/views/parts/index.html.erb

index 33ff8e0..bbc1cf9 100644

--- a/app/views/parts/index.html.erb

+++ b/app/views/parts/index.html.erb

@@ -25,9 +25,3 @@

<br>

<%= link\_to 'New Part', new\_part\_path %>

-<%= form\_tag(search\_parts\_url, method: "get") do %>

- <%= label\_tag(:search, "Search by Part Name:") %>

- <%= text\_field\_tag :search, params[:search],

- data: { autocomplete: autocomplete\_part\_name\_parts\_path } %>

- <%= submit\_tag("Search") %>

-<% end %>

# Route Update

diff --git a/config/routes.rb b/config/routes.rb

index c38fcdb..44136c2 100644

--- a/config/routes.rb

+++ b/config/routes.rb

@@ -1,29 +1,6 @@

Rails.application.routes.draw do

- root 'cars#index'

-

- resources :parts do

- get :autocomplete\_part\_name, on: :collection

-

- collection do

- get 'search'

- end

- end

-

- resources :makes do

- get :autocomplete\_make\_name, on: :collection

-

- collection do

- get 'search'

- end

- end

-

- resources :cars do

- get :autocomplete\_car\_model, on: :collection

-

- collection do

- get 'search'

- end

- end

-

+ resources :cars

+ resources :parts

+ resources :makes

# For details on the DSL available within this file, see http://guides.rubyonrails.org/routing.html

end

# Testing Updates

## Car Search Tests (updated creation test with a unique VIN)

diff --git a/test/controllers/cars\_controller\_test.rb b/test/controllers/cars\_controller\_test.rb

index 7a36bac..c2f842f 100644

--- a/test/controllers/cars\_controller\_test.rb

+++ b/test/controllers/cars\_controller\_test.rb

@@ -3,27 +3,8 @@ require 'test\_helper'

class CarsControllerTest < ActionDispatch::IntegrationTest

setup do

@car = cars(:one)

- # @update = {

- # make\_id: makes(:one),

- # model: "my model",

- # vin: 200

- # }

end

- # Custom Tests

-

- test "should find a car that exists" do

- get search\_cars\_url, params: {search: "MyString"}

- assert\_select 'td', 'MyString'

- end

-

- test "shouldn't find a car that doesnt exist" do

- get search\_cars\_url, params: {search: "I dont exist"}

- assert\_select 'td', false

- end

-

- # Scaffold Tests

-

test "should get index" do

get cars\_url

assert\_response :success

@@ -36,24 +17,12 @@ class CarsControllerTest < ActionDispatch::IntegrationTest

test "should create car" do

assert\_difference('Car.count') do

- post cars\_url, params: { car: { make\_id: @car.make\_id, model: @car.model, vin: 4 } }

+ post cars\_url, params: { car: { make\_id: @car.make\_id, model: @car.model, vin: @car.vin } }

end

assert\_redirected\_to car\_url(Car.last)

end

- # test "shouldn't create a car if the make does not exist" do

- # assert\_no\_difference('Car.count') do

- # post cars\_url, params: { car: { make\_id: "not a valid make", model: @car.model, vin: 4 } }

- # end

- #

- # # assert\_redirected\_to car\_url(Car.last)

- #

- # # post car

- # # car = Car.create(model: "model", vin: 400, make: "not a valid make")

- # # refute car.valid?

- # end

-

test "should show car" do

get car\_url(@car)

assert\_response :success

## Make search tests

diff --git a/test/controllers/makes\_controller\_test.rb b/test/controllers/makes\_controller\_test.rb

index 93a22b3..52dfd39 100644

--- a/test/controllers/makes\_controller\_test.rb

+++ b/test/controllers/makes\_controller\_test.rb

@@ -5,18 +5,6 @@ class MakesControllerTest < ActionDispatch::IntegrationTest

@make = makes(:one)

end

- test "should find a make that exists" do

- get search\_makes\_url, params: {search: "MyString"}

- assert\_select 'td', 'MyString'

- end

-

- test "shouldn't find a make that doesnt exist" do

- get search\_makes\_url, params: {search: "I dont exist"}

- assert\_select 'td', false

- end

-

- # Scaffold Tests

-

test "should get index" do

get makes\_url

assert\_response :success

## Parts search tests

diff --git a/test/controllers/parts\_controller\_test.rb b/test/controllers/parts\_controller\_test.rb

index 8c89059..0191216 100644

--- a/test/controllers/parts\_controller\_test.rb

+++ b/test/controllers/parts\_controller\_test.rb

@@ -5,18 +5,6 @@ class PartsControllerTest < ActionDispatch::IntegrationTest

@part = parts(:one)

end

- test "should find a part that exists" do

- get search\_parts\_url, params: {search: "MyString"}

- assert\_select 'td', 'MyString'

- end

-

- test "shouldn't find a part that doesnt exist" do

- get search\_parts\_url, params: {search: "I dont exist"}

- assert\_select 'td', false

- end

-

- # Scaffold Tests

-

test "should get index" do

get parts\_url

assert\_response :success

## Updated fixture config to match db schema (uniqueness)

diff --git a/test/fixtures/cars.yml b/test/fixtures/cars.yml

index e93d513..0e67f35 100644

--- a/test/fixtures/cars.yml

+++ b/test/fixtures/cars.yml

@@ -8,4 +8,4 @@ one:

two:

make: two

model: MyString

- vin: 2

+ vin: 1

# Model Tests

## Car validation tests

diff --git a/test/models/car\_test.rb b/test/models/car\_test.rb

index 65ff56f..39bdaec 100644

--- a/test/models/car\_test.rb

+++ b/test/models/car\_test.rb

@@ -1,45 +1,7 @@

require 'test\_helper'

class CarTest < ActiveSupport::TestCase

- test "should create a car with a unique VIN and all fields present" do

- car = Car.create(model: "model", vin: 400, make: makes(:one))

- assert car.valid?

- assert car.save

- end

-

- test "should not allow duplicate VINs" do

- car = Car.create(model: "model", vin: cars(:one).vin, make: makes(:one))

- refute car.valid?

-

- assert\_equal car.errors.messages, {:vin=>["has already been taken"]}

- end

-

- test "shouldn't allow a non-number VIN" do

- car = Car.create(model: "model", vin: "not a number", make: makes(:one))

- refute car.valid?

-

- assert\_equal car.errors.messages, {:vin=>["is not a number"]}

- end

-

- test "should require a VIN" do

- car = Car.create(model: "model", make: makes(:one))

- refute car.valid?

-

- assert\_equal car.errors.messages, {:vin=>["can't be blank", "is not a number"]}

- end

-

- test "should require a model" do

- car = Car.create(make: makes(:one), vin: 400)

- refute car.valid?

-

- assert\_equal car.errors.messages, {:model=>["can't be blank"]}

- end

-

- test "should require a make" do

- car = Car.create(model: "model", vin: 400)

- refute car.valid?

-

- assert\_equal car.errors.messages, {:make=>["must exist"], :make\_id=>["can't be blank"]}

- end

-

+ # test "the truth" do

+ # assert true

+ # end

end

## Make validation tests

diff --git a/test/models/make\_test.rb b/test/models/make\_test.rb

index 26c3100..8162f47 100644

--- a/test/models/make\_test.rb

+++ b/test/models/make\_test.rb

@@ -1,25 +1,7 @@

require 'test\_helper'

class MakeTest < ActiveSupport::TestCase

- test "shouldn't be valid without a name" do

- make = Make.create(country: "my country")

- refute make.valid?

-

-

- assert\_equal make.errors.messages, {:name=>["can't be blank"]}

- end

-

- test "shouldn't be valid without a country" do

- make = Make.create(name: "my name")

- refute make.valid?

-

-

- assert\_equal make.errors.messages, {:country=>["can't be blank"]}

- end

-

- test "should be valid with a name and a country" do

- make = Make.create(name: "my name", country: "my country")

- assert make.valid?

- assert make.save

- end

+ # test "the truth" do

+ # assert true

+ # end

end

## Part validation tests

diff --git a/test/models/part\_test.rb b/test/models/part\_test.rb

index d23f06a..bba3103 100644

--- a/test/models/part\_test.rb

+++ b/test/models/part\_test.rb

@@ -1,13 +1,7 @@

require 'test\_helper'

class PartTest < ActiveSupport::TestCase

- test "should require a name" do

- part = Part.new

- refute part.valid?

- assert\_equal part.errors.messages, {:name=>["can't be blank"]}

-

- part.name = "my name"

- assert part.valid?

- assert part.save

- end

+ # test "the truth" do

+ # assert true

+ # end

end

# Other Updates

## Javascript to enable autocomplete

diff --git a/app/assets/javascripts/cars.coffee b/app/assets/javascripts/cars.coffee

index 59713bf..24f83d1 100644

--- a/app/assets/javascripts/cars.coffee

+++ b/app/assets/javascripts/cars.coffee

@@ -1,9 +1,3 @@

# Place all the behaviors and hooks related to the matching controller here.

# All this logic will automatically be available in application.js.

# You can use CoffeeScript in this file: http://coffeescript.org/

-

-$(document).on 'turbolinks:load', ->

- $("input[data-autocomplete]").each ->

- url = $(this).data('autocomplete')

- $(this).autocomplete

- source: url

diff --git a/app/assets/javascripts/makes.coffee b/app/assets/javascripts/makes.coffee

index 59713bf..24f83d1 100644

--- a/app/assets/javascripts/makes.coffee

+++ b/app/assets/javascripts/makes.coffee

@@ -1,9 +1,3 @@

# Place all the behaviors and hooks related to the matching controller here.

# All this logic will automatically be available in application.js.

# You can use CoffeeScript in this file: http://coffeescript.org/

-

-$(document).on 'turbolinks:load', ->

- $("input[data-autocomplete]").each ->

- url = $(this).data('autocomplete')

- $(this).autocomplete

- source: url

diff --git a/app/assets/javascripts/parts.coffee b/app/assets/javascripts/parts.coffee

index 59713bf..24f83d1 100644

--- a/app/assets/javascripts/parts.coffee

+++ b/app/assets/javascripts/parts.coffee

@@ -1,9 +1,3 @@

# Place all the behaviors and hooks related to the matching controller here.

# All this logic will automatically be available in application.js.

# You can use CoffeeScript in this file: http://coffeescript.org/

-

-$(document).on 'turbolinks:load', ->

- $("input[data-autocomplete]").each ->

- url = $(this).data('autocomplete')

- $(this).autocomplete

- source: url

## CSS requirement update

diff --git a/app/assets/stylesheets/application.css b/app/assets/stylesheets/application.css

index 89e3819..d05ea0f 100644

--- a/app/assets/stylesheets/application.css

+++ b/app/assets/stylesheets/application.css

@@ -12,5 +12,4 @@

\*

\*= require\_tree .

\*= require\_self

- \*= require jquery-ui

\*/

## Javascript Requirement update

diff --git a/app/assets/javascripts/application.js b/app/assets/javascripts/application.js

index 533bfc7..82e6f0f 100644

--- a/app/assets/javascripts/application.js

+++ b/app/assets/javascripts/application.js

@@ -11,7 +11,6 @@

// about supported directives.

//

//= require rails-ujs

+//= require activestorage

//= require turbolinks

-//= require jquery

-//= require jquery-ui

//= require\_tree .

## Gem File Updates for AutoComplete and JQuery

diff --git a/Gemfile b/Gemfile

index 62959bb..19ae71f 100644

--- a/Gemfile

+++ b/Gemfile

@@ -60,7 +60,3 @@ end

# Windows does not include zoneinfo files, so bundle the tzinfo-data gem

gem 'tzinfo-data', platforms: [:mingw, :mswin, :x64\_mingw, :jruby]

-

-gem 'autocomplete\_rails'

-gem 'jquery-rails'

-gem 'jquery-ui-rails'

diff --git a/Gemfile.lock b/Gemfile.lock

index 9e98df1..e08aa7b 100644

--- a/Gemfile.lock

+++ b/Gemfile.lock

@@ -47,8 +47,6 @@ GEM

archive-zip (0.12.0)

io-like (~> 0.3.0)

arel (9.0.0)

- autocomplete\_rails (0.4.1)

- rails (>= 4.0, < 5.3)

bindex (0.8.1)

bootsnap (1.4.5)

msgpack (~> 1.0)

@@ -86,12 +84,6 @@ GEM

io-like (0.3.0)

jbuilder (2.9.1)

activesupport (>= 4.2.0)

- jquery-rails (4.3.5)

- rails-dom-testing (>= 1, < 3)

- railties (>= 4.2.0)

- thor (>= 0.14, < 2.0)

- jquery-ui-rails (6.0.1)

- railties (>= 3.2.16)

listen (3.1.5)

rb-fsevent (~> 0.9, >= 0.9.4)

rb-inotify (~> 0.9, >= 0.9.7)

@@ -199,15 +191,12 @@ PLATFORMS

ruby

DEPENDENCIES

- autocomplete\_rails

bootsnap (>= 1.1.0)

byebug

capybara (>= 2.15)

chromedriver-helper

coffee-rails (~> 4.2)

jbuilder (~> 2.5)

- jquery-rails

- jquery-ui-rails

listen (>= 3.0.5, < 3.2)

puma (~> 3.11)

rails (~> 5.2.3)

## GitConfig update

diff --git a/.gitignore b/.gitignore

index f3ae195..81452db 100644

--- a/.gitignore

+++ b/.gitignore

@@ -4,8 +4,6 @@

# or operating system, you probably want to add a global ignore instead:

# git config --global core.excludesfile '~/.gitignore\_global'

-.idea/

-

# Ignore bundler config.

/.bundle