

Updating Search & Adding Show All Button

To add a before/after graduation date to your search field, you will need to update your `student_controller.rb` to handle the additional data as well as update your `search_form.html.erb` file to have the drop down you need.

Updating Student Controller

Here's what your index method should look like in your `student_controller.rb`:

```
# GET /students or /students.json
def index
  @search_params = params[:search] || {}

  if @search_params.empty?
    @students = []
  else
    @students = Student.all

    # Filter by Major
    if @search_params[:major].present? && @search_params[:major] != "Any Major"
      @students = @students.where(major: @search_params[:major])
    end

    # Filter by Graduation Date
    if @search_params[:date_filter].present? && @search_params[:graduation_date].present?
      graduation_date = Date.parse(@search_params[:graduation_date]) rescue nil

      if graduation_date
        if @search_params[:date_filter] == 'before'
          @students = @students.where('graduation_date < ?', graduation_date)
        elsif @search_params[:date_filter] == 'after'
          @students = @students.where('graduation_date > ?', graduation_date)
        end
      end
    end
  end
end
```

(Figure 1)

Updated Student Controller Breakdown

The additional if statement for graduation date and date filter layer on top of the previous if statement so that they happen together. Additionally it handles the grad date base on if you select before or after.

Updating Search Form

The basic `search_form.html.erb` file you have should look something like Figure 2, you may not have as many `<div>` elements as I do, but you will when you bootstrap so don't even sweat it.

```
<%= form_with(url: students_path, method: :get, local: true) do |form| %>
  <div>
    <%= form.label :major, "Major" %>
    <%= form.select "search[major]", options_for_select(Student::VALID_MAJORS, @search_params[:major]),
      prompt: "Any Major" %>
  </div>

  <div>
    <%= form.label :date_filter, "Before/After" %>
    <%= form.select "search[date_filter]", options_for_select(['Before', 'before'], ['After', 'after'],
      @search_params[:date_filter]), prompt: "Before/After" %>
    <%= form.label :graduation_date, "Graduation" %>
    <%= form.date_field "search[graduation_date]", value: @search_params[:graduation_date] %>
  </div>

  <div>
    <%= form.submit "Search" %>
    <%= link_to "Show All", students_path(search: { major: "" }) %>
    <%= link_to "New student", new_student_path %>
  </div>
<% end %>
```

(Figure 2)

Updated Search Form Breakdown

Additional form labels and fields have been added for the `date_filter` and the `graduation_date`.

Bootstrap your Search Form

I bootstrapped my search form in two ways. Firstly in the `search_form.html.erb` file itself (Figure 3), making the form elements more modern. Secondly, in the `index.html.erb` (Figure 4) file to display the search form within a card element.

```
<%= form_with(url: students_path, method: :get, local: true, class: "form-inline") do |form| %>
  <div class="form-group d-flex align-items-center mb-2">
    <%= form.label :major, "Major", class: "form-label mt-2 me-2" %>
    <%= form.select "search[major]", options_for_select(Student::VALID_MAJORS, @search_params[:major]),
      { prompt: "Any Major" }, class: "form-control" %>
  </div>

  <div class="form-group d-flex align-items-center mb-2">
    <%= form.label :date_filter, "Before/After", class: "form-label mt-2 me-2" %>
    <%= form.select "search[date_filter]", options_for_select([['Before', 'before'], ['After', 'after']],
      @search_params[:date_filter]), { prompt: "Before/After" }, class: "form-control" %>
    <%= form.label :graduation_date, "Graduation", class: "form-label ms-2 mt-2 me-2" %>
    <%= form.date_field "search[graduation_date]", value: @search_params[:graduation_date], class: "form-control" %>
  </div>

  <div class="form-group d-flex justify-content-end">
    <%= form.submit "Search", class: "btn btn-primary me-2" %>
    <%= link_to "Show All", students_path(search: { major: "" }), class: "btn btn-secondary me-2" %>
    <%= link_to "New student", new_student_path, class: "btn btn-secondary" %>
  </div>
<% end %>
```

(Figure 3)

Bootstrapped Search Form Breakdown

My form is sectioned off into `<div>` elements so that bootstrap class attributes can be added that align them and center them. **“d-flex”** is a type of container that is dynamic for different screens, but allows me to get my labels and form fields on the same line.

Various margins have been added, **“me-2”** and **“mt-2”**, for example, which add margins to the end and top respectively.

The buttons at the bottom also received bootstrap **“btn”** classes, which help the buttons stand out.

There are many ways to organize your forms, I would recommend checking out the bootstrap official site for more information.

Bootstrap Your Index For Search

You're probably currently just rendering `search_form`, but I chose to place my search form into a `<div>` container and turn it into a "card" element to help it stand out from the page. (Figure 4)

Again, lots of ways to parse off your search form, so I would recommend exploring the web.

```
<div class="row">
  <div class="col-md-8 mx-auto">
    <div class="card shadow-sm">
      <div class="card-body">
        <%= render 'search_form', search_params: @search_params %>
      </div>
    </div>
  </div>
</div>
```

(Figure 4)

ARIA Labels

Copy your search form into ChatGPT and ask it to add ARIA labels where it makes sense.