Active record Associations

Original Author: Dr. Diego Martinez, TA, 2021-2022 AY

https://guides.rubyonrails.org/association_basics.html

Goal:

Create a new CRUD in your Book Collection app that interacts with a 'join table' (i.e., associative entity) which results from two entities having a many to many relationships.

Steps (selected):

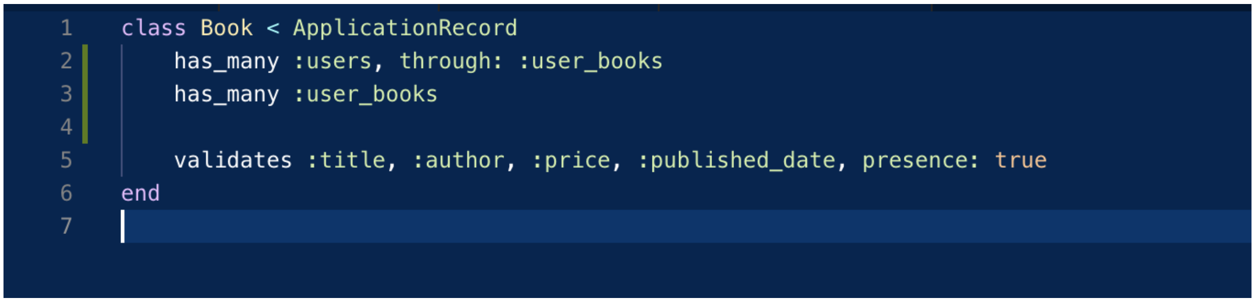
1. Create a new entity using scaffold, the new entity is User with a single attribute username:string
2. Modify the attributes to represent many to many users to books
3. Create a new entity (i.e., associative entity) that contains user_id:integer and book_id:integer (hint: scaffold is a great tool)
4. Modify the new view (titled "User Books") on the new entity to show only the available users and books.
5. Add links to the book and users indexes.
6. Modify the routes file so the root route goes to the new view mentioned in the previous item.
7. Push to the Github classroom, using link below:
   Book Collection 4

Hints :

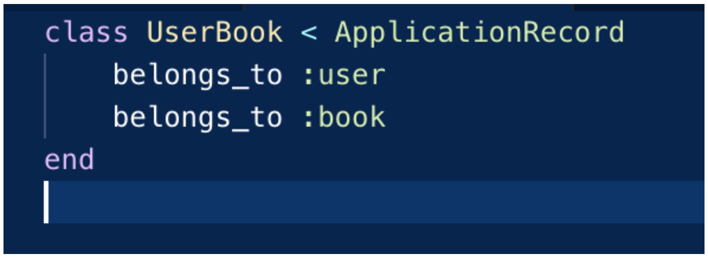
- Create a new entity using scaffold, the new entity is User with a single attribute username:string

  ```
  rails g scaffold User username:string

  rails g scaffold UserBook user_id:integer  book_id:integer
  ```

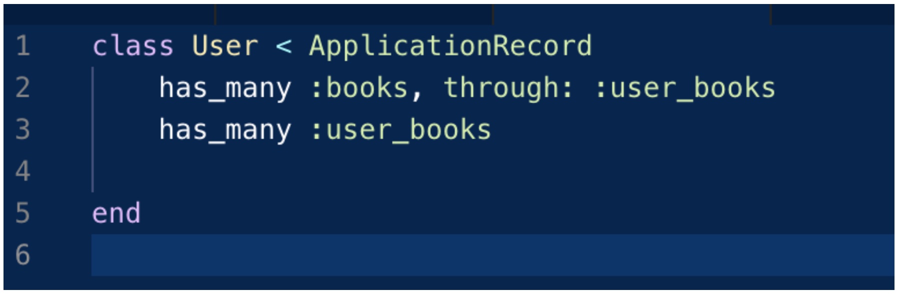
- Modify the attributes to represent many to many users to books



```ruby
class Book < ApplicationRecord
    has_many :users, through: :user_books
    has_many :user_books

    validates :title, :author, :price, :published_date, presence: true
end
```



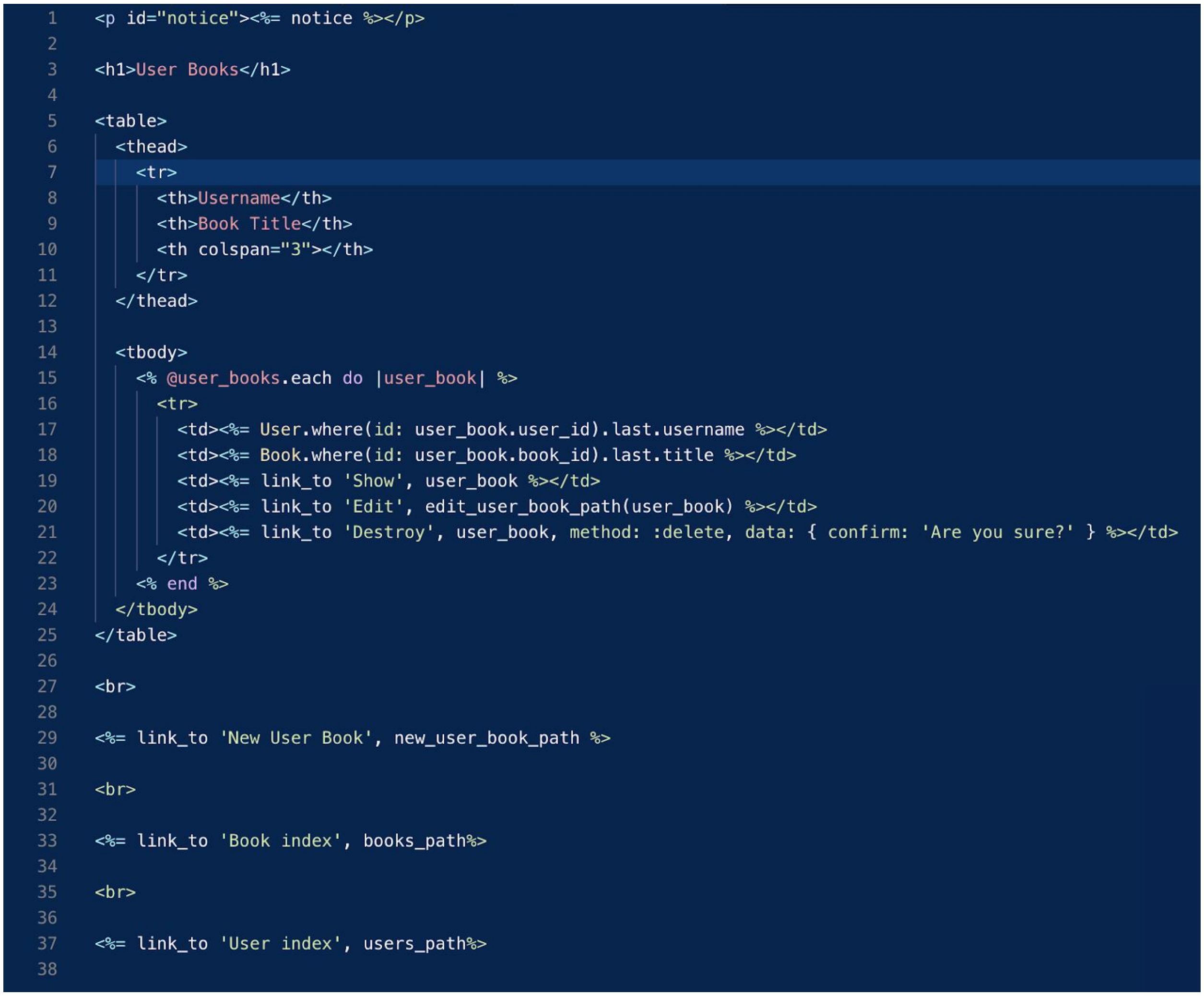
```ruby
class UserBook < ApplicationRecord
    belongs_to :user
    belongs_to :book
end
```



```ruby
class User < ApplicationRecord
    has_many :books, through: :user_books
    has_many :user_books

end
```

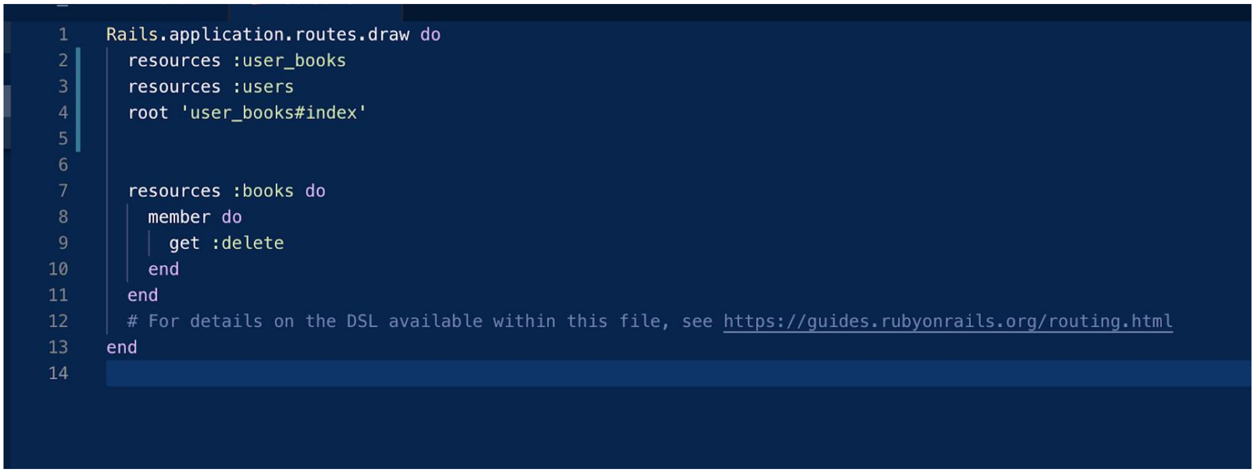
- Create a new entity that contains user_id:integer and book_id:integer (hint: scaffold is a great tool) **(done above)**

- Modify the new view on the new entity to show only the available users and books.
- Add links to the book and users indexes.



```
<p id="notice"><%= notice %></p>

<h1>User Books</h1>

<table>
  <thead>
    <tr>
      <th>Username</th>
      <th>Book Title</th>
      <th colspan="3"></th>
    </tr>
  </thead>

  <tbody>
    <% @user_books.each do |user_book| %>
      <tr>
        <td><%= User.where(id: user_book.user_id).last.username %></td>
        <td><%= Book.where(id: user_book.book_id).last.title %></td>
        <td><%= link_to 'Show', user_book %></td>
        <td><%= link_to 'Edit', edit_user_book_path(user_book) %></td>
        <td><%= link_to 'Destroy', user_book, method: :delete, data: { confirm: 'Are you sure?' } %></td>
      </tr>
    <% end %>
  </tbody>
</table>

<br>

<%= link_to 'New User Book', new_user_book_path %>

<br>

<%= link_to 'Book index', books_path%>

<br>

<%= link_to 'User index', users_path%>

```

- Modify the routes file so the root route goes to the new view



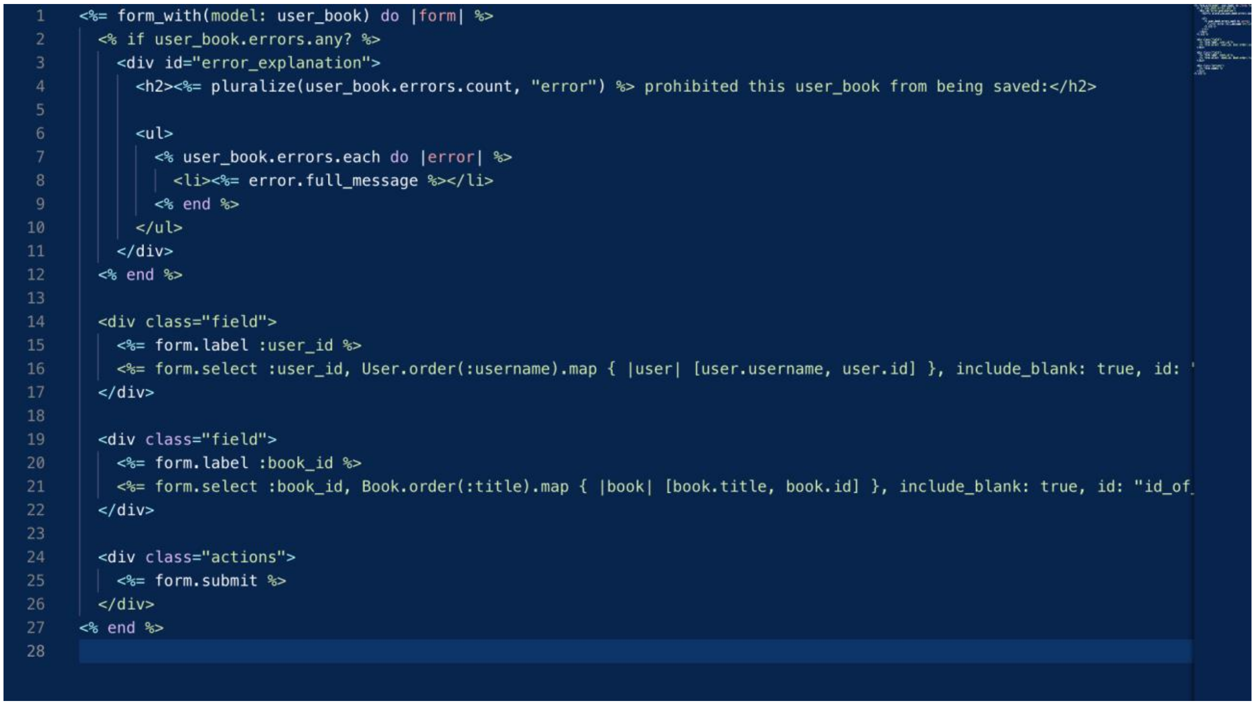
```
Rails.application.routes.draw do
  resources :user_books
  resources :users
  root 'user_books#index'


  resources :books do
    member do
      get :delete
    end
  end
  # For details on the DSL available within this file, see https://guides.rubyonrails.org/routing.html
end

```

You need to add

<%= form.select :user_id, User.order(:username).map { |user| [user.username, user.id] }, include_blank: true, id: "id_of_the_user_id_select_box" %>

(shown in the screenshot below)

In app/views/user_books/_form.html.erb and also you need to edit one more form field in the same file identify and do. (you need to do this so that localhost:3000/user_books/new has dropdown list of all available users and books)



```
<%= form_with(model: user_book) do |form| %>
  <% if user_book.errors.any? %>
    <div id="error_explanation">
      <h2><%= pluralize(user_book.errors.count, "error") %> prohibited this user_book from being saved:</h2>

      <ul>
        <% user_book.errors.each do |error| %>
          <li><%= error.full_message %></li>
        <% end %>
      </ul>
    </div>
  <% end %>

  <div class="field">
    <%= form.label :user_id %>
    <%= form.select :user_id, User.order(:username).map { |user| [user.username, user.id] }, include_blank: true, id: '
  </div>

  <div class="field">
    <%= form.label :book_id %>
    <%= form.select :book_id, Book.order(:title).map { |book| [book.title, book.id] }, include_blank: true, id: "id_of_
  </div>

  <div class="actions">
    <%= form.submit %>
  </div>
<% end %>

```

```
   <%= form.select :user_id, User.order(:username).map { |user| [user.username,
user.id] }, include_blank: true, id: "id_of_the_user_id_select_box" %>
```

```
   <%= form.select :book_id, Book.order(:title).map { |book| [book.title, book.id] },
include_blank: true, id: "id_of_the_book_id_select_box" %>
```

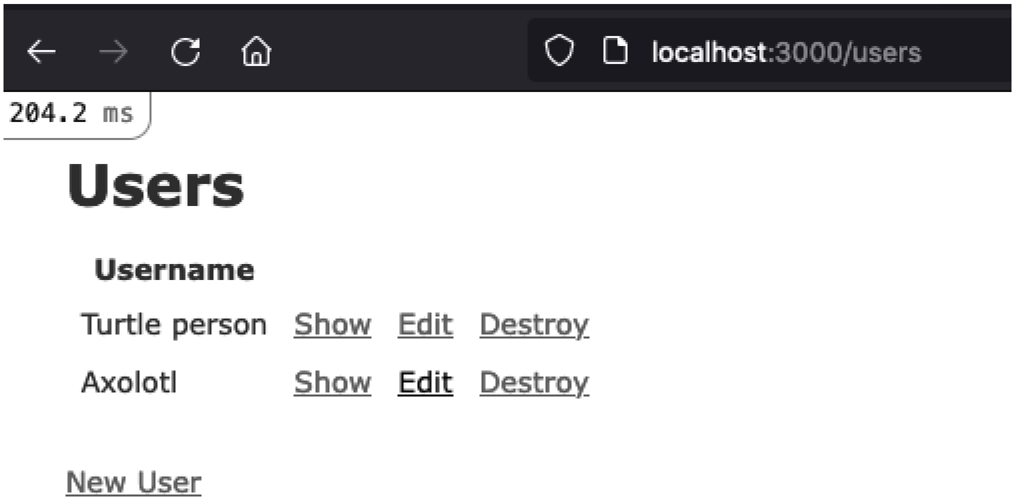
Submission:

- Provide the screenshot of the user view (1)
- Provide the screenshot of creating a new user_book instance (2)
- Provide the screenshot of the new view (titled “User Books”) which shows only the available users and books, with links to book index, user index and new user_book. (3)
- Provide the screenshot of the routes file, showing user_book index as the root route
- Provide screenshots of merging changes into the proper Github branches
-Provide screenshots of commit history
-Submit a screenshot of an analysis of your code against the Canvas rubric using the LLM of your choice. This should communicate improvement suggestions to comply with the given rubric.

Save your report as FirstName_LastName_Lab-BookColl2ection4.pdf and submit it to Canvas.

Final views should look like this :

1.)

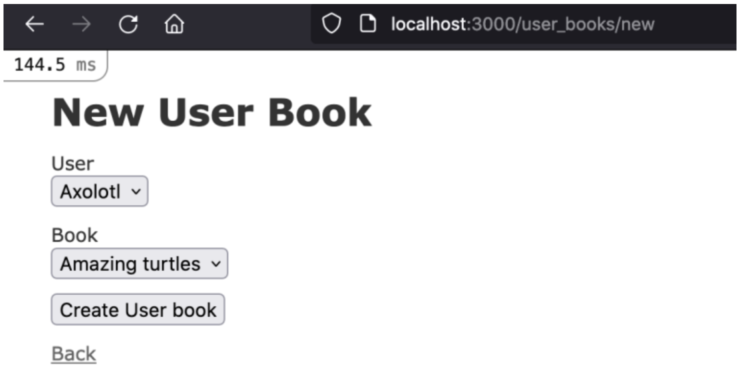


Users

Username			
Turtle person	Show	Edit	Destroy
Axolotl	Show	Edit	Destroy

New User

2.)



localhost:3000/user_books/new

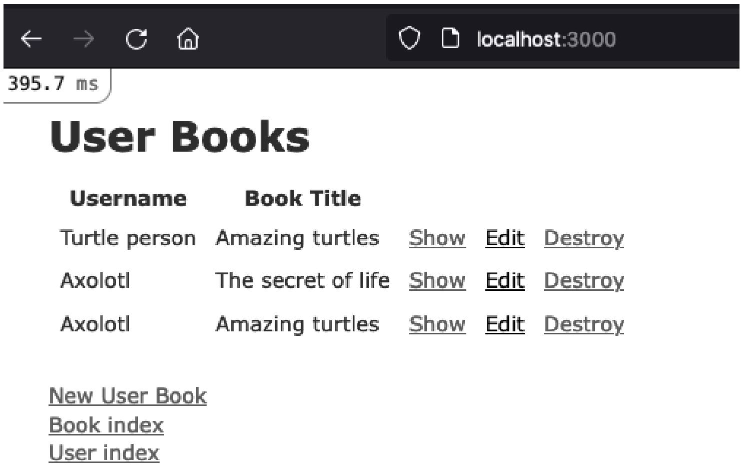
144.5 ms

New User Book

User
Axolotl

Book
Amazing turtles

3.)



localhost:3000

395.7 ms

User Books

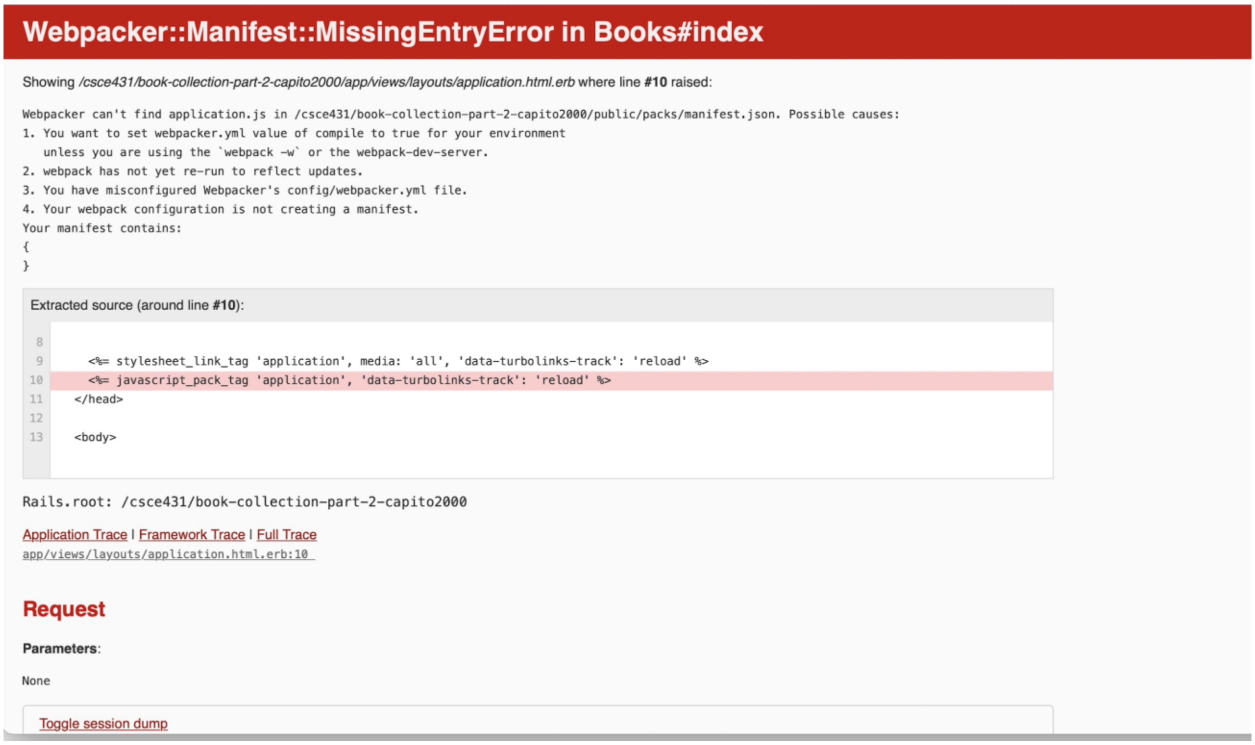
Username	Book Title			
Turtle person	Amazing turtles	Show	Edit	Destroy
Axolotl	The secret of life	Show	Edit	Destroy
Axolotl	Amazing turtles	Show	Edit	Destroy

Grading

Will be graded using the following sections of the Sprint Rubric:

- **Data Design**

If you are facing the below error, you can try running this command :



Webpacker::Manifest::MissingEntryError in Books#index

Showing */csce431/book-collection-part-2-capito2000/app/views/layouts/application.html.erb* where line **#10** raised:

```
Webpacker can't find application.js in /csce431/book-collection-part-2-capito2000/public/packs/manifest.json. Possible causes:
1. You want to set webpacker.yml value of compile to true for your environment
   unless you are using the `webpack -w` or the webpack-dev-server.
2. webpack has not yet re-run to reflect updates.
3. You have misconfigured Webpacker's config/webpacker.yml file.
4. Your webpack configuration is not creating a manifest.
Your manifest contains:
{
}
```

Extracted source (around line **#10**):

```
8
9       <%= stylesheet_link_tag 'application', media: 'all', 'data-turbolinks-track': 'reload' %>
10      <%= javascript_pack_tag 'application', 'data-turbolinks-track': 'reload' %>
11    </head>
12
13    <body>
```

```
Rails.root: /csce431/book-collection-part-2-capito2000
```

Application Trace | Framework Trace | Full Trace

app/views/layouts/application.html.erb:10

Request

Parameters:

None

Toggle session dump

bundle exec rake webpacker:install