

```
rails new myapp
rails server
rails generate controller books index
rails generate model Book
rails db:migrate
rails generate migration AddDescriptionToBooks
```

rails console

```
Book.connection
Book
Book.create(title: 'example')
Book.first
Book.last
Book.find(1)
Book.find_by_title('example')
Book.all
Book.select(:title)
Book.where(title: 'example')
b = Book.new
b = Book.where(title: 'example')
b.title = 'other example'
b.save
Book.last.destroy
```

```
def edit
  @title = "Buch bearbeiten"
  @action = "update"
  @book = Book.find(params[:id])
  @subjects = Subject.all
end

def update
  @book = Book.find(params[:id])
  if @book.update(book_params)
    redirect_to action: "index"
  else
    @subjects = Subject.all
    render action: "edit"
  end
end

def destroy
  Book.find(params[:id]).destroy
  redirect_to action: "index"
end
```

```
Rails.application.routes.draw do
  get '/users/show'
  root 'pages#home'
  get 'helloworld', to: 'helloworld#index'
  get '/pages/texte'
  get '/pages/bilder'
  get '/pages/formulare'

  get '/subjects', to: 'subjects#index', as: 'subjects'
  get '/subjects/show/:id', to: 'subjects#show', as: 'show_subject'

  get '/books', to: 'books#index', as: 'books'
  get '/books/show/:id', to: 'books#show', as: 'show_book'
  delete '/books/destroy/:id', to: 'books#destroy', as: 'delete_book'
  get '/books/new', to: 'books#new', as: 'new_book'
  post '/books', to: 'books#create'
  get '/books/edit/:id', to: 'books#edit', as: 'edit_book'
  patch '/books/:id', to: 'books#update'

  resources :albums
  resources :songs
end
```

```
<%= form_with model: @book do |f| %>
  <p>Titel:</p>
  <%= f.text_field :title %>
  <p>Preis:</p>
  <%= f.text_field :price %>
  <p>Subject:</p>
  <%= f.collection_select :subject_id, @subjects, :id, :name %>
  <p>Description:</p></td>
  <%= f.text_area :description %>
  <%= submit_tag "Buch speichern" %>
  <%= link_to "Abbrechen", books_path %>
<% end %>
```

```
<% @books.each do |b| %>
  <%= link_to b.title, show_book_path(b.id) %>
  <%= link_to "Ändern", edit_book_path(b.id) %>
  <%= link_to "Löschen", delete_book_path(b.id), method: :delete, data: { confirm: "Sind Sie sicher?" } %>
<% end %>
```

- ✓ app
 - > assets
 - > channels
 - > controllers
 - > helpers
 - > javascript
 - > jobs
 - > mailers
 - > models
 - ✓ views
 - > albums
 - > books
 - > helloworld
 - > layouts
 - > pages
 - > songs
 - > subjects
 - > users
 - > bin
 - > config
 - ✓ db
 - > migrate
 - development.sqlite3
 - schema.rb
 - seeds.rb

HTTP Verb	Pfad	Controller#Action	Beschreibung
GET	/albums	albums#index	Liste mit allen Alben anzeigen
GET	/albums/new	albums#new	HTML-Formular anzeigen zum Erfassen eines neuen Albums
POST	/albums	albums#create	Neues Album in die DB speichern
GET	/albums/:id	albums#show	Details eines Albums anzeigen
GET	/albums/:id/edit	albums#edit	HTML-Formular anzeigen zum Bearbeiten eines Albums
PATCH/PUT	/albums/:id	albums#update	Änderungen in die DB speichern
DELETE	/albums/:id	albums#destroy	Ein Album löschen

```
def book_params
  params.require(:book).permit(:title, :price, :subject_id, :description)
end
```

```
def index
  @books = Book.all
end

def show
  @book = Book.find(params[:id])
end

def new
  @title = "Buch erfassen"
  @action = "create"
  @book = Book.new
  @subjects = Subject.all
  render action: "edit"
end

def create
  @book = Book.new(book_params)
  if @book.save
    redirect_to action: "index"
  else
    @subjects = Subject.all
    render action: "new"
  end
end
```

Helper	Gibt zurück
albums_path	/albums
new_album_path	/albums/new
edit_album_path(:id)	/albums/:id/edit
album_path(:id)	/albums/:id

```
class CreateBooks < ActiveRecord::Migration[6.0]
  def change
    create_table :books do |t|
      t.string :title
      t.float :price
      t.integer :subject_id
      t.timestamps
    end
  end
end
```

```
class Subject < ApplicationRecord
  has_many :books, dependent: :restrict_with_error
end
```

```
class Book < ApplicationRecord
  belongs_to :subject
  validates_presence_of :title, message: "darf nicht leer sein"
  validates_numericality_of :price, :message => "darf nur eine Zahl sein"
end
```