Ruby on Rails Cheat Sheet

by Akshaya (akki_1) via cheatography.com/208452/cs/44671/

What is Ruby on rails?

- It is a server side web application framework written in ruby

- Follow MVC Architecture & provides default structures for databases, web pages, web services.

Migration

- It is a feature in rails that allow developer to manage & modify database schema over time.

- There are Ruby files to be changes to be made to database(Creating tables, adding columns)

- Making easier to track changes, rollback changes if necessary

Gemfile

It is used to manage the applications ruby gem dependencies. It lists all the gems that app needs & bundle us used to install them.

MVC Architecture

Model (M): Represents the data and business logic. It handles database interactions, validations, and any logic that pertains to the data.

View (V): Manages what the user sees. It is responsible for displaying data provided by the model in a formatted way using HTML, CSS, and embedded Ruby (ERB).

Controller (C): Acts as an intermediary between the Model and View. It handles user requests, interacts with the model to get or update data, and sends the appropriate response back to the view for display.

Flow: User -> Controller -> Model -> Controller -> View -> User.

Active Record

It provides methods for CRUD operations, associations beterrn models, validations and more, making it easier to work with database in rails app.

ORM(Object-Relational-Mapping)

It allows developers to interact with database using ruby code instead of writing raw SQL queries.

Routing

Routing connects incoming requests (URLs) to the correct controller actions. Rails uses a routes.rb file to define routes.

A route can be defined as: get 'articles', to: 'articles#index'

- a GET request to /articles will be handled by the index action in the ArticlesController.

- Rails has RESTful routes, which map standard URL patterns to controller actions (e.g., index, show, create, etc.) based on resources:

resources :articles

- This generates routes for all CRUD actions (index, show, new, edit, create, update, destroy).

Callbacks

Callbacks are methods that get triggered at certain points in the lifecycle of an object (like before saving, after updating, etc.).

Common callbacks include:
before_save: Runs before saving a record.

after_save: Runs after saving a record.

before_validation: Runs before validation happens.

after_create: Runs after a new record is created.

They are used to execute code at specific points to maintain data consistency or add additional logic.

render vs redirect_to

render: It view template without making a new request(redirecting to browser). Simply display view..

redirect_to: Redirect browser to another action/ URL, triggering new request.

find vs find_by vs where

Find: It looks up a record using its primary key(id). It gives error if records isn't found.

find_by: Searches for first record that matches certain condition, return nil if no match found.

where: Finds all records that meet specific condition & returns them as list of objects.

resource vs resources

resource: It will give routes to all command operations except Index.

resources: It will give routes to all command operations

validate vs validates

validate : It is used to define custom validations method.

validates : It is used to apply built-in validation rules to model attributes.