



D04 - Formation Ruby on Rails

English version coming soon !

Résumé: Rails un framework opensource puissant aliant productivité élevée et qualité de code, tout en bénéficiant d'une grande communauté passionnée et très réactive. Avec ce D04, vous allez en domestiquer les parties les plus abordables.

Table des matières

I	Préambule	2
II	Consignes	3
III	Règles spécifiques de la journée	5
IV	Exercice 00 : CheatSheet	6
V	Exercice 01 : Moar CheatSheet	8
VI	Exercice 02 : Quick search	10
VII	Exercice 03 : Diary	11

Chapitre I

Préambule

La philosophie

- DRY : Don't Repeat Yourself
- Convention over configuration
- REST : Representational State Transfer
- CRUD : Create, Read, Update, Delete

La bibliothèque

- [Votre nouvelle bible](#)
- [Votre nouvelle boîte à outil](#)
- [Votre nouveau JT](#)
- [Votre nouvelle TV](#)

Chapitre II

Consignes

- Seule cette page servira de référence : ne vous fiez pas aux bruits de couloir.
- Le sujet peut changer jusqu'à une heure avant le rendu.
- Si aucune information contraire n'est explicitement présente, vous devez assumer les versions de langages suivantes :
 - Ruby `>= 2.3.0`
 - pour d09 Rails `> 5`
 - mais pour tous les autres jours Rails `4.2.7`
 - HTML 5
 - CSS 3
- Nous vous interdisons FORMELLEMENT d'utiliser les mots clés `while`, `for`, `redo`, `break`, `retry`, `loop` et `until` dans les codes sources Ruby que vous rendrez. Toute utilisation de ces mots clés est considérée comme triche (et/ou impropre), vous donnant la note de -42.
- Les exercices sont très précisément ordonnés du plus simple au plus complexe. En aucun cas, nous ne porterons attention ni ne prendrons en compte un exercice complexe si un exercice plus simple n'est pas parfaitement réussi.
- Attention aux droits de vos fichiers et de vos répertoires.
- Vous devez suivre la procédure de rendu pour tous vos exercices : seul le travail présent sur votre dépôt GIT sera évalué en soutenance.
- Vos exercices seront évalués par vos camarades de piscine.
- Vous ne devez laisser dans votre répertoire aucun autre fichier que ceux explicitement spécifiés par les énoncés des exercices.
- Vous avez une question ? Demandez à votre voisin de droite. Sinon, essayez avec votre voisin de gauche.
- Votre manuel de référence s'appelle `Google` / `man` / `Internet` /
- Pensez à discuter sur le forum Piscine de votre Intra, ou Slack, ou IRC...

- Lisez attentivement les exemples. Ils pourraient bien requérir des choses qui ne sont pas autrement précisées dans le sujet...
- Par pitié, par Thor et par Odin ! Réfléchissez nom d'une pipe !

Chapitre III

Règles spécifiques de la journée

- Tous les fichiers rendus seront des applications web Ruby On Rails.
- Tout projet de la journée doit inclure dans son Gemfile :


```
gem 'rubycritic', :require => false
```

Lors de l'exécution de la commande "rubycritic", la section "smells" de la page html de rapport DOIT ETRE VIDE!!

- Sauf mention du sujet, AUCUNE Gem supplémentaire n'est autorisée pour accomplir ces exercices.

Chapitre IV

Exercice 00 : CheatSheet

	Exercice : 00
Exercice 00 : CheatSheet	
Dossier de rendu : <i>ex00/</i>	
Fichiers à rendre : CheatSheet	
Fonctions Autorisées : bootstrap-sass , rubycritic	

Non content d'avoir fait hier votre premier "Hello World" en rails , vous vous sentez pousser des ailes et decidez d'aller plus loin dans l'utilisation de votre nouveau compagnon.

Pas trop loin cependant... En effet, nous allons toute la journée exploiter rails dans ses mécanismes les plus simples.

Pour le coup, si Wikipedia a occupé la majorité de votre temps hier, je vous conseille de faire le D03-Ex04 avant de continuer.

PAS DE BASE DE DONNEES NI DE 'CRUD' pour l'instant, nous allons nous concentrer sur les vues.

Rendez-vous [ici](#) et contemplez ce merveilleux site de rappels de commandes utiles...

Vous devez reproduire cette page, footer compris. On parle que de la page hein, le système de navigation, ce sera pour plus tard. D'ailleurs, perdez pas de temps à faire du css au pixel près, cependant un peu de mise en page et une certaine ressemblance est demandée.

Ces quelques points sont à respecter :

- L'app s'appellera "CheatSheet"
- Elle ne contient qu'un controller de votre création , `application_controller` qui est à laisser tel quel
- La page principale aura pour title "CheatSheet"
- La nav-bar (le truc rouge, là, en début de page) n'est pas à reproduire pour le moment
- Un peu de mise en page est de mise. Nous ne vous demandons pas un design digne d'une app à mettre en production, mais d'une mise en page simple et agréable à regarder.


Votre HTML doit être là où il est censé être, le controller doit être dans son dossier. Aucun CSS ne doit être présent dans un fichier HTML ! Servez vous de Rails comme il se doit.

D'ailleurs, ne soyez pas bêtes, la CheatSheet porte bien son nom et vous aidera beaucoup dans l'arborescence d'une application Rails ainsi que des commandes utiles.

Servez vous aussi de la commande "inspect element" de votre navigateur et soyez malins, rappelez vous le crawler d'hier.

Chapitre V

Exercice 01 : Moar CheatSheet

	Exercice : 01
Exercice 01 : Moar CheatSheet	
Dossier de rendu : <i>ex01/</i>	
Fichiers à rendre : NewCheatSheet	
Fonctions Autorisées : bootstrap-sass, rubycritic	

Bon, maintenant que vous savez faire une One-Page avec Rails, il est temps de parler sérieux maintenant.

Reprenez votre précédente application. Vous allez maintenant implémenter la nav-bar (truc rouge, tout ca...) et changer un peu la navigation de votre CheatSheet.

Concrètement, vous allez séparer chaque partie de cette CheatSheet en pages indépendantes, et assigner un bouton de la nav-bar à la page correspondante.


Ce qu'on attends de vous :

- L'app s'appellera "NewCheatSheet"
- Elle ne contient toujours qu'un controller de votre création , **application_controller** qui est à laisser tel quel
- Chaque page a sa balise "title" dédiée
- Vous ne devez écrire le code de la "nav-bar" qu'une seule fois!
- La première page est la page "Convention", ce sera votre root-page

- Toutes les pages suivantes doivent être représentées :
 - convention
 - console
 - ruby
 - ruby-concepts
 - ruby-numbers
 - ruby-strings
 - ruby-arrays
 - ruby-hashes
 - rails-folder-structure
 - rails-commands
 - rails-erb
 - editor
 - help
- Les pages "Resources" et "About" ne sont pas à reproduire, ni à inclure dans la nav-bar.
- Un peu de mise en page est de mise. Nous ne vous demandons pas un design digne d'une app à mettre en production, mais d'une mise en page simple et agréable à regarder.

Chapitre VI

Exercice 02 : Quick search

	Exercice : 02
Exercice 02 : Quick search	
Dossier de rendu : <i>ex02/</i>	
Fichiers à rendre : CheatSheet	
Fonctions Autorisées : bootstrap-sass, rubycritic, jquery-datatables-rails	

Plutôt pratique cette CheatSheet, mais elle n'est pas encore au top.

Vous allez l'améliorer encore avec un plugin **jQuery** et en reformatant un peu le contenu... Ajoutez un onglet nommé "Quick search" qui va contenir un tableau qui incorpore le contenu de TOUS les autres tableaux de la CheatSheet.

Ce tableau sera **correctement formé**, aura notamment un header, le bon nombre de colonnes, des balises bien placées etc etc.

Ainsi vous pourrez utiliser la gem **jquery-datatables-rails** qui va vous dynamiser ce tableau.


En clair vous devez :

- Implémenter un nouvel onglet "Quick search" dans la nav-bar.
- Faire en sorte que cet onglet ait sa balise title personnalisée, comme le reste du site.
- Faire en sorte que son contenu soit un tableau récapitulatif de TOUTES les commandes des autres pages.
- Associer le plugin **jquery-datatables-rails** à ce tableau, et l'inclure dans le projet via sa Gem.

NB : Si un popup d'erreur s'affiche ou si du Javascript est présent dans le fichier `html.erb`, l'exercice sera compté faux.

Chapitre VII

Exercice 03 : Diary

	Exercice : 03
Exercice 03 : Diary	
Dossier de rendu : <i>ex03/</i>	
Fichiers à rendre : CheatSheet	
Fonctions Autorisées : bootstrap-sass, rubycritic, jquery-datatables-rails	

Le projet CheatSheet commence à avoir un certain aplomb. Mais comme tout bon développeur, vous tenez un journal technique, et l'incorporer à votre projet serait un atout non négligeable.

Créez un nouvel onglet nommé "Log Book" où figure en tête de page, un formulaire avec un bouton 'Write' et un champ pour inclure du texte. Une fois le formulaire renseigné avec du texte et soumis, votre app ajoutera le texte du formulaire à un fichier "entry_log.txt", placé à la racine de votre dossier de projet CheatSheet

Le texte devra y être formaté à la volée en ajoutant en début de ligne, la date et l'heure comme ceci :

```
15/07/2016 23:42:04 : Tout l'equipage est endormi, Jim.  
15/07/2016 23:41:26 : Tout l'equipage baille, Jim.
```

L'affichage du texte est fait avec une boucle : chaque entrée doit être affichée à la suite du formulaire, avec sa propre balise et ce, de l'entrée la plus récente à l'entrée la plus ancienne.

A chaque fois qu'on clique sur le bouton "Write" du formulaire, vous devez faire en sorte d'être redirigé sur la même page avec la nouvelle ligne affichée en haut.