



Initiation à la programmation en Ruby - 42

Jour 00

Staff 42 bocal@42.fr

Résumé: Ce document est le sujet du jour 00 de la piscine d'initiation à la programmation en Ruby.

Table des matières

I	Consignes	2
II	Exercice 00 : La base	3
III	Exercice 01 : WIFILLES	4
IV	Exercice 02 : can't touch this	5
V	Exercice 03 : retour vers le futur	6
VI	Exercice 04 : testDay00	7
VII	Exercice 05 : Lister l'arborescence	9
VIII	Exercice 06 : Parse it baby !	11


Chapitre I

Consignes

- A 42, vous allez faire l'expérience d'une pédagogie un peu particulière : vous avez un "cours" d'introduction d'1h tous les matins, et le reste de la journée, vous avez des exercices à réaliser en autonomie.
- Vous avez une question ? Un problème ? Un blocage ? Demandez à votre voisine de droite. Sinon, essayez avec votre voisine de gauche. A 42, les étudiants ne sont pas en compétition, ils avancent ensemble, en s'entre-aidant.
- Votre manuel de référence s'appelle **Google / man / Internet /** Vous allez devoir apprendre à faire des recherches sur internet, toutes les infos dont vous avez besoin s'y trouvent !
- Lisez attentivement les exemples. Vous devez respecter le formatage des réponses : les majuscules, les retours à la ligne... tout est important. Programmer, c'est avant tout faire preuve de rigueur.
- Un tuteur vous accompagne tout au long de la piscine : il/elle est là pour vous donner des pistes, vous indiquer comment faire vos recherches sur internet et vous encourager si vous êtes démotivées. Le tuteur est un soutien pour vous, mais il n'est pas là pour vous donner les réponses !
- A la fin de la journée, le tuteur de votre rangée va corriger votre travail collectivement. C'est un bon moment pour échanger et s'expliquer, entre élèves, les erreurs que vous avez pu faire, ou au contraire expliquer aux autres ce que vous avez compris. Soyez attentives.
- Bon courage, et n'ayez pas peur de vous tromper ! Faites des tests, tatonnez - en informatique, c'est en faisant des erreurs qu'on apprend !

Chapitre II

Exercice 00 : La base

	Exercice : 00
On va partir sur de bonnes bases	
Dossier de rendu : <code>ex00/</code>	
Fichiers à rendre : aucun	
Fonctions Autorisées : Aucune	
Remarques : n/a	

- Créez un dossier à la racine de votre home, il s'appellera `piscine_octobre`.
- Déplacez vous à l'intérieur de ce dossier, et créez un nouveau dossier `jour00`.
- De nouveau, déplacez vous dans ce dossier, et créez-y un nouveau dossier `ex00`. Ne mettez rien dedans.
- A partir de maintenant, tous les exercices suivants devront être dans le bon dossier de rendu. L'exercice 01 dans le dossier `ex01`, l'exercice 02 dans le dossier `ex02`, etc... vous avez compris la logique.




Attention, `man` = manuel. Faites "`man mkdir`" et "`man cd`" dans votre shell, vous aurez la page de manuel qui s'affiche.
Les manuels sont en anglais. Si vous ne parlez pas du tout anglais, vous pouvez chercher sur Google, des traductions existent.



Vous devez réaliser cet exercice avec `mkdir` et `cd`, sinon ça n'a aucun intérêt :)

Chapitre III

Exercice 01 : WIFILLES

	Exercice : 01
Les vraies savent afficher WiFilles	
Dossier de rendu : <i>ex01/</i>	
Fichiers à rendre : wifilles	
Fonctions Autorisées : Aucune	
Remarques : n/a	

- Créez un fichier **wifilles**.
- Il doit être dans le bon répertoire de rendu, comme demandé dans l'exercice précédent.
- Lorsque l'on fait un **cat** dessus, cela doit afficher "WiFilles" suivi d'un retour à la ligne.


```
?> cat wifilles
WiFilles
?>
```



Faites "man cat" dans votre shell.

Chapitre IV

Exercice 02 : can't touch this

	Exercice : 02
Est-ce que tu as les droits ?	
Dossier de rendu : <i>ex02/</i>	
Fichiers à rendre : cant_touch_this	
Fonctions Autorisées : Aucune	
Remarques : n/a	

- Créez un fichier `cant_touch_this`.
- Vous devez modifier les droits de votre fichier, pour obtenir les droits suivants :

```
?> ls -l
total 1
-r-x----- 1 laurie  staff      0 Oct  8 16:53 cant_touch_this
?>
```




Cherchez sur internet pour comprendre ce que sont les droits d'un fichier. Faites également un `man ls` et un `man chmod`.



Les colonnes "laurie" et "staff" seront remplacés respectivement par votre login et votre groupe.

Chapitre V

Exercice 03 : retour vers le futur

	Exercice : 03
Et si on faisait des trucs dans le futur ?	
Dossier de rendu : <i>ex03/</i>	
Fichiers à rendre : <i>in_the_futur</i>	
Fonctions Autorisées : Aucune	
Remarques : n/a	

- Créez un fichier *in_the_futur*.
- Modifiez le pour que lorsqu'on fait un *ls* dessus, la date soit celle de Noël à venir :


```
?> ls -l
total 1
-rw-r--r--  1 laurie  staff           0 Dec 25  2015 in_the_futur
?>
```



Faites man touch

Chapitre VI

Exercice 04 : testDay00

	Exercice : 04
Modifier plein d'attributs	
Dossier de rendu : <i>ex04/</i>	
Fichiers à rendre : testDay00.tar	
Fonctions Autorisées : Aucune	
Remarques : n/a	

- Créez le fichier **testDay00** dans votre répertoire de rendu.
- Vous devez reproduire les permissions, la taille, et la date, comme ci-dessous :

```
?> ls -l
total 1
-r--r-xr-x 1 login wheel 40 Feb 1 23:42 testDay00
?>
```

- Une fois l'exercice résolu, vous exécuterez la commande ci-dessous pour créer le fichier d'archive **testDay00.tar** (il n'y pas de piège, faites la commande telle quelle) :

```
?> tar -cf testDay00.tar testDay00
?>
```

- Supprimez maintenant le fichier **testDay00** de votre répertoire. Il ne doit rester que le fichier **testDay00.tar** dans le répertoire.




Faites man rm.



Les colonnes "login" et "wheel" seront remplacés respectivement par votre login et votre groupe.

Chapitre VII

Exercice 05 : Lister l'arborescence

	Exercice : 05
Lister l'arborescence	
Dossier de rendu : <i>ex05/</i>	
Fichiers à rendre : exo.sh	
Fonctions Autorisées : Aucune	
Remarques : n/a	

- Créer un fichier **exo.sh**.
- Il doit contenir une commande permettant de lister récursivement tous les fichiers et dossiers du répertoire parent.
- Si vous avez fait correctement les exercices précédents, vous obtiendrez la sortie suivante :

```
?> ./exo.sh
ex00    ex01    ex02    ex03    ex04    ex05

../ex00:

../ex01:
wifilles

../ex02:
cant_touch_this

../ex03:
in_the_futur

../ex04:
testDay00.tar

../ex05:
exo.sh
?>
```



Commencez par chercher ce qu'est un répertoire parent.




Comment faire pour lister récursivement des fichiers? Il existe une option pour ça : faites un `man ls`!



Pour être exécutable, votre fichier doit avoir les bonnes permissions. Pour rappel : `man chmod`.

Chapitre VIII

Exercice 06 : Parse it baby !

	Exercice : 06
Let's parse it !	
Dossier de rendu : <i>ex06/</i>	
Fichiers à rendre : <code>parse_it_baby.sh</code>	
Fonctions Autorisées : Aucune	
Remarques : n/a	

- Créez un fichier `parse_it_baby.sh`
- Il doit contenir une ligne de commande. Cette commande va lire le fichier `listing_students.csv` qui comprend des informations sous la forme `NOM;Prénom;email;ÉCOLE` et va en extraire le mail. Vous ne récupérerez que la partie avant le `@` de l'adresse email.
- Vous trouverez le fichier `listing_students.csv` en annexe sur l'intranet 42.
- La sortie attendue :

```
?> ./parse_it_baby.sh | cat -e
quentin.bollache$
laurie.mezard$
andre.aubin$
[...]
mathieu.mahe$
frederic.grati$
?>
```



Faites man cut. Vous allez aussi utiliser de quoi lire un fichier, vous l'avez fait au début de la journée.



La commande `ls -la | cut -f 1 -d" "` se fait en deux temps, la seconde partie utilise le résultat de la première.

Google shell pipe pour comprendre de quoi il s'agit, et comment vous pouvez l'utiliser.