

# Chapitre 0 : Plan

## INF3080 BASES DE DONNÉES (SGBD)

Guy Francoeur

Aucune reproduction sans autorisation

3 septembre 2019

**UQÀM** | **Département d'informatique**

# Table des matières

1. Présentation du cours
2. Environnement Linux (Unix)
3. Rappel des commandes utiles
4. Retour sur les logiciels d'éditeurs
5. Le logiciel Git (rappel)

# Table des matières

1. Présentation du cours
2. Environnement Linux (Unix)
3. Rappel des commandes utiles
4. Retour sur les logiciels d'éditeurs
5. Le logiciel Git (rappel)

# Informations générales

Trimestre	Automne 2019
Titre du cours	Base de données
Sigle	INF3080
Département	Informatique
Enseignant	Guy Francoeur, PK-4115
Coordonnateur	Fatiha Sadat, professeur
Lien utile	<a href="https://etudier.uqam.ca/cours?sigle=INF3080">https://etudier.uqam.ca/cours?sigle=INF3080</a>
Page du cours	<a href="https://github.com/guyfrancoeur/INF3080">https://github.com/guyfrancoeur/INF3080</a>

# Description du cours

- ▶ Voir plan de cours.

# Modalités d'évaluations proposées

- ▶ **Examens** :

- ▶ Examen **intra** [30%];

- ▶ Examen **final** [30%];

- ▶ **Travaux pratiques** :

- ▶ TP1 : Conception [15%];

- ▶ TP2 : Construction et usage [25%];

- ▶ **Retard, absence et plagiat** : voir politiques UQAM.

- ▶ Elles seront mises à jour au besoin dans la page du cours.

# Table des matières

1. Présentation du cours
2. Environnement Linux (Unix)
3. Rappel des commandes utiles
4. Retour sur les logiciels d'éditions
5. Le logiciel Git (rappel)



- ▶ Cette session nous allons travailler dans environnement Linux/Unix.
- ▶ Votre CodeMS est requis.
- ▶ Vérifier que vous êtes en mesure de vous connecter sur **zeta2.labunix.uqam.ca** (via putty, ssh ou terminal)
- ▶ Quel est le port par défaut pour une connexion ssh?

- ▶ Dans ce cours, nous travaillerons avec le **terminal** ou la **console**.
- ▶ Le terminal agit comme **intermédiaire** pour lancer vos commandes dans le système d'exploitation **Linux**;
- ▶ SQL\*plus est un produit/outil © Oracle;
- ▶ SQL\*plus est lancé dans un terminal en ligne de commande;
- ▶ SQL\*plus est un terminal pour **discuter** avec le SGBDR © Oracle.

# Table des matières

1. Présentation du cours
2. Environnement Linux (Unix)
3. Rappel des commandes utiles
4. Retour sur les logiciels d'éditions
5. Le logiciel Git (rappel)

# Commandes agissant sur les fichiers

- ▶ Résumé (**cheatsheet**) :

<https://ubuntudanmark.dk/filer/fwunixref.pdf>

- ▶ Commandes **fréquentes** :

Commande	Description
<code>ls</code>	Liste les fichiers dans un répertoire
<code>ls -alhs</code>	Liste tout (-a) détaillée (-l) humain (-h) size -s
<code>cd</code>	Change de répertoire
<code>mkdir</code>	Crée un répertoire
<code>rm</code>	Supprime un fichier (irréversible)
<code>rm -rf</code>	Supprime un répertoire (récursivement)
<code>cp</code>	Copie un fichier
<code>cp -r</code>	Copie un dossier (récursivement)
<code>mv</code>	Renomme/déplace un fichier/répertoire

## Autres commandes de fichiers

Commande	Description
<code>find</code>	Trouve toutes les occurrences d'un fichier
<code>cat</code>	Affiche le contenu d'un fichier ou concatène plusieurs fichiers
<code>less</code>	Permet de parcourir le contenu d'un fichier
<code>head</code>	Affiche les premières lignes d'un fichier
<code>tail</code>	Affiche les dernières lignes d'un fichier
<code>pwd</code>	Affiche le répertoire courant
<code>touch</code>	Crée un fichier vide ou s'il existe, modifie sa date au moment présent
<code>file</code>	Affiche l'encodage ASCII, UTF-8, Unix ...
<code>iconv</code>	Convertir le codepage
<code>wget</code>	Télécharge un fichier
<code>curl</code>	Télécharge un fichier

Commande	Description
<code>top</code>	Affichage interactif des processus actifs
<code>ps</code>	Affiche les processus actifs
<code>kill</code>	Tue un processus (fin de tâche)
<code>chmod</code>	Change les permissions d'un fichier ou un dossier
<code>which</code>	Affiche le chemin d'une application
<code>grep</code>	Recherche une expression régulière
<code>date</code>	Affiche l'heure et la date
<code>time</code>	Affiche le temps d'exécution requis par un processus

# Table des matières

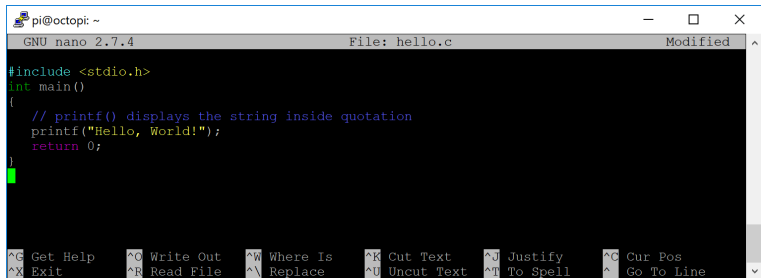
1. Présentation du cours
2. Environnement Linux (Unix)
3. Rappel des commandes utiles
4. Retour sur les logiciels d'éditions
5. Le logiciel Git (rappel)

- ▶ Vous avez le choix entre **Vim** et **Nano**;
- ▶ J'utiliserai **Nano** principalement pour les exemples;
- ▶ Dans tous les cas, assurez-vous que vos fichiers sont enregistrés au format **ISO-8859** ou **ascii**;
- ▶ Tout fichier ayant un problème d'**encodage** sera considéré comme non valide;
- ▶ **\*\*** Sauvegarde de fichiers UTF-8 avec l'option **UTF8 with noBOM**.



- ▶ Un très simple **éditeur de texte**;
- ▶ La première version de nano a été écrite par Chris Allegretta en 1999;
- ▶ Son ancêtre est **pico**, qui n'est pas gratuit;
- ▶ La configuration de GNU nano se fait à l'aide du fichier **.nanorc**;
- ▶ Les fichiers de configuration sont dans **/usr/share/nano/**
- ▶ `$ cd ; cat /usr/share/nano/c.nanorc >> .nanorc`

# GNU nano



The screenshot shows a terminal window with the GNU nano 2.7.4 text editor. The editor is editing a file named 'hello.c'. The code visible in the editor is:

```
#include <stdio.h>
int main()
{
    // printf() displays the string inside quotation
    printf("Hello, World!");
    return 0;
}
```

The bottom status bar of the nano editor displays various keyboard shortcuts for navigation and editing, such as ^G Get Help, ^O Write Out, ^W Where Is, ^K Cut Text, ^J Justify, ^C Cur Pos, ^X Exit, ^R Read File, ^\ Replace, ^U Uncut Text, ^T To Spell, and ^\_ Go To Line.

- ▶ **ctrl** + **x** sortir (demande pour la sauvegarde);
- ▶ **ctrl** + **o** sauvegarder;
- ▶ **ctrl** + **w** rechercher dans le fichier;
- ▶ **ctrl** + **k** effacement de la ligne au complet.

# Table des matières

1. Présentation du cours
2. Environnement Linux (Unix)
3. Rappel des commandes utiles
4. Retour sur les logiciels d'éditeurs
5. Le logiciel Git (rappel)

# Logiciel de gestion de versions

- ▶ Permet de **conserver** plusieurs **fichiers** et plusieurs projets aussi appelés **repository**;
- ▶ Conserve l'historique et la chronologie de toutes les modifications effectuées;
- ▶ Offre des services de **collaboration** entre plusieurs **participants**;
- ▶ Permet également de gérer différentes **branches** dont les évolutions sont temporairement **indépendantes**.
- ▶ Garantit une certaine tranquillité, car il est toujours possible de **revenir en arrière**;
- ▶ Git gère la sécurité de vos projets privés et les accès sont limités aux collaborateurs;
- ▶ Nous utilisons le gestionnaire de version Git ici.

# Commandes les plus courantes

Quelques opérations **courantes** de Git :

- ▶ **Créer** un nouveau projet : `git init`;
- ▶ **Cloner** un projet existant : `git clone`;
- ▶ **Sauvegarder** l'état courant du projet : `git commit`;
- ▶ **Versionner** un nouveau fichier : `git add`;
- ▶ **Ajouter** un fichier pour le prochain commit : `git add`;
- ▶ **Consulter** l'historique : `git log`;
- ▶ **Récupérer** des changements à distance : `git pull`;
- ▶ **Téléverser** des changements à distance : `git push`, etc.

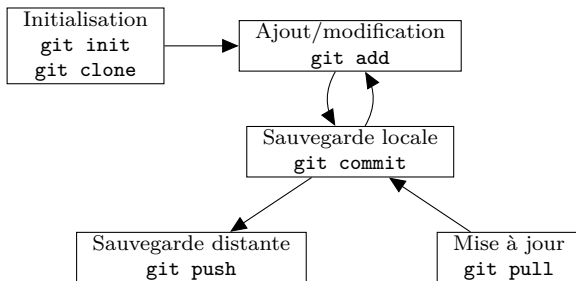
# Configuration de Git

- ▶ La configuration de Git est **très simple**;
- ▶ La configuration est gardée dans un **fichier texte** nommé **.gitconfig** généralement stocké dans le dossier **\$HOME**;
- ▶ Le fichier sera créé et rempli grâce à certaines commandes disponibles. Ceci est plus simple qu'éditer le fichier manuellement;
- ▶ Voici les commandes pour certaines configurations :

```
$ git config --global user.name "username"
$ git config --global user.email "email@domaine.ext"
$ git config --global core.editor nano
$ git config --global color.ui auto
$ git config --global push.default simple
```

# Flux opérationnel (*workflow*)

Aide mémoire Git:



# Conclusion

À la fin, vous serez en mesure :

- ▶ de différencier les différents types de gestionnaires de BD;
- ▶ de comprendre la modélisation conceptuelle et relationnelle;
- ▶ de construire et utiliser des modèles ou schémas de données;
- ▶ de connaître et utiliser langage de requêtes SQL;
- ▶ d'utiliser et manipuler des bases de données avec SQL;
- ▶ construire des programmes en PL/SQL;
- ▶ écrire des requêtes efficaces et/ou optimiser des requêtes SQL;
- ▶ connaître et connecter une application à une base de données;