

ANTOINE LECACHEUR

ANALYSTE DÉVELOPPEUR AU SNDI - ORLÉANS

- Mail pro : antoine.lecacheur@insee.fr
- Mail perso : antoine.lecacheur@hotmail.fr

DÉCOUPAGE EN COUCHES

- Couche de présentation, IHM (User Interface)
- Couche métier (Business Logic)
- La couche d'accès aux données (Data Access)

COUCHE DE PRÉSENTATION

- Permet l'interaction entre un utilisateur et votre application
- Regroupe tout ce qui à trait à l'affichage
- **En général** : une page web, une application mobile...
- **Dans le cadre des projets informatiques** : c'est la console

COUCHE MÉTIER

- Aussi appelée couche 'logique' de l'application
- Crée et manipule des objets
- Effectue les traitements sur ces objets
- Fait appel à des [web services](#)
- **Exemple** : pour twitter un objet métier essentiel est le tweet.
- Il se caractérise par son contenu (dont la taille est limitée), est associé à des commentaires, etc.

COUCHE DAO

- Data Access Object
- Répond à la question : **Comment stocker nos données "métiers" ?**
- Classe technique DAO - Persiste les données d'une classe métier
- Permet en particulier de faire le lien avec une base de données
- Expose donc les méthodes **CRUD** : ajouter, lire, modifier ou supprimer des données

L'INTÉRÊT DES COUCHES

TP1 - DAO

NOUVEL IDE : PYCHARM

- Mieux adapté que Spyder pour le développement d'application
- De nombreux autres IDE existent, Visual Studio Code est également disponible pour les projets infos

UNE LIBRAIRIE POUR GÉRER LES CONNEXIONS À LA BDD : [PSYCOPG2](#)

PETIT POINT SUR L'INJECTION SQL

- ```
SELECT * from admins WHERE login='admin' AND password='$password'
```
- Imaginons qu'on envoie, pour password la valeur : " 'OR 1=1 -- "
- ```
SELECT * from admins WHERE login='admin' AND password='' OR 1=1
```
- L'utilisateur arrive alors à s'authentifier et accéder à l'application !

DIFFÉRENTS MOYENS D'EMPÊCHER L'INJECTION SQL

- Échapper les caractères spéciaux
- Utiliser une requête préparée
- psycopg2 s'occupe d'échapper les caractères spéciaux pour nous !

RÉCUPÉRER LE TP : INTRODUCTION À GIT

- Logiciel de gestion de versions décentralisé
- Gérer un dépôt de fichiers partagé par plusieurs personnes
- Chaque ajout de fichiers (**commit**) est enregistrée en tant que nouvelle version, on conserve donc l'historique de tous les changements
- Gérer les conflits lorsque plusieurs personnes travaillent sur le même projet

UN REPOSITORY GIT ?

- Repository = dépôt
- Repository local et repository distant
- Chaque ajout de fichiers (**commit**) est enregistrée en tant que nouvelle version, on conserve donc l'historique de tous les changements
- Gérer les conflits lorsque plusieurs personnes travaillent sur le même projet

LES COMMANDES GIT POUR LE TP VIA GIT BASH

PLACE AU TP!