

Programmation Python

***Mini-Projet Python***

***« DB2File »***

Master Spécialisé Big Data & Cloud Computing

Imane JABAL

Mohamed AIT ABDERRAHMAN

Rachid REBIK

Encadré par:

M. Noreddine ABGHOUR

Année Universitaire 2015/2016

SOmmaire

[1 Introduction 2](#_Toc441918279)

[2 Modules utilisés 2](#_Toc441918280)

[3 Test 2](#_Toc441918281)

[4 Conclusion 2](#_Toc441918282)

# Introduction

De nombreux problèmes peuvent être exprimés en termes de contraintes comme les problèmes d’emploi de temps, de gestion d’agenda, de gestion de trafic ainsi que certains problèmes de planification et d’optimisation (comme le problème de routage de réseaux de télécommunication). On divisera cette étude en 3 parties principales :

# Problématique

Notre problématique consiste de générer des classes à partir d’une base de données. Ensuite stockées ces classes dans un fichier binaire.

Après on va convertir notre fichier binaire vers un format bien précis (json, xml…).

# Réalisation

## Outils

**PyCharm :**

PyCharm est un IDE spécialisé pour Python/Django.

C'est gratuit ? PyCharm existe en deux versions : une gratuit & open-source (Free Community) et une autre propriétaire et payante (Full-fledged Professional, 99$ pour un an de license).  
La version Free Community suffit amplement  
  
Les avantages ? PyCharm dispose d'une interface sobre et moderne, très agréable !  
Quelques fonctionnalités :

* Auto-complétion du code (même par des variables d'un module importé !)
* Vérification du code
* Vérification PEP8
* Aide à la réalisation des docstrings
* Suggestions de correction automatique/optimisation du code
* Gestion efficace de projets
* Débuggeur intégré
* etc...

**Python :**

Python est un [langage de programmation](https://fr.wikipedia.org/wiki/Langage_de_programmation) [objet](https://fr.wikipedia.org/wiki/Programmation_orient%C3%A9e_objet), multi-[paradigme](https://fr.wikipedia.org/wiki/Paradigme_(programmation)) et [multiplateformes](https://fr.wikipedia.org/wiki/Plate-forme_(informatique)). Il favorise la [programmation impérative](https://fr.wikipedia.org/wiki/Programmation_imp%C3%A9rative) [structurée](https://fr.wikipedia.org/wiki/Programmation_structur%C3%A9e), [fonctionnelle](https://fr.wikipedia.org/wiki/Programmation_fonctionnelle) et [orientée objet](https://fr.wikipedia.org/wiki/Programmation_orient%C3%A9e_objet). Il est doté d'un [typage dynamique](https://fr.wikipedia.org/wiki/Typage_dynamique) [fort](https://fr.wikipedia.org/wiki/Typage_fort), d'une gestion automatique de la mémoire par [ramasse-miettes](https://fr.wikipedia.org/wiki/Ramasse-miettes_(informatique)) et d'un [système de gestion d'exceptions](https://fr.wikipedia.org/wiki/Syst%C3%A8me_de_gestion_d%27exceptions).

**MySQL :**

MySQL est un [système de gestion de bases de données](https://fr.wikipedia.org/wiki/Syst%C3%A8me_de_gestion_de_base_de_donn%C3%A9es) relationnelles (SGBDR). Il est distribué sous une double licence [GPL](https://fr.wikipedia.org/wiki/Licence_publique_g%C3%A9n%C3%A9rale_GNU) et [propriétaire](https://fr.wikipedia.org/wiki/Logiciel_propri%C3%A9taire). Il fait partie des logiciels de gestion de [base de données](https://fr.wikipedia.org/wiki/Base_de_donn%C3%A9es) les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec [Oracle](https://fr.wikipedia.org/wiki/Oracle_Database), [Informix](https://fr.wikipedia.org/wiki/Informix) et [Microsoft SQL Server](https://fr.wikipedia.org/wiki/Microsoft_SQL_Server).

## Modules utilisés

**Sqlacodegen :**

Automatic model code generator for SQLAlchemy : est module de génération du code automatique pour SQLAlchemy.

**SQLAlchemy :**

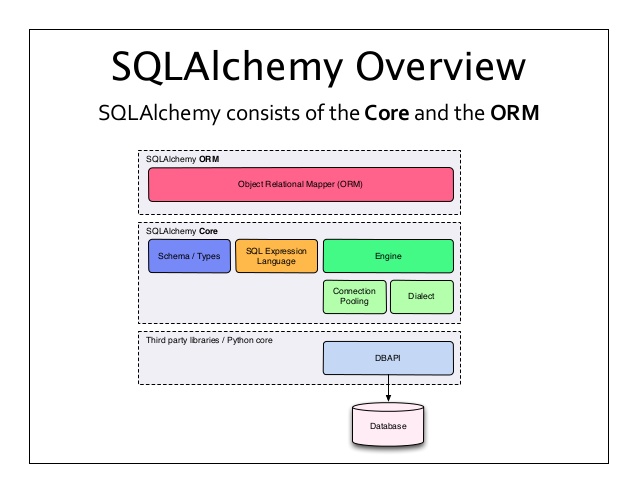
****

SQLAlchemy est un toolkit [open source](https://fr.wikipedia.org/wiki/Open_source) [SQL](https://fr.wikipedia.org/wiki/Structured_Query_Language) et un [mapping objet-relationnel](https://fr.wikipedia.org/wiki/Mapping_objet-relationnel) (ORM) écrit en [Python](https://fr.wikipedia.org/wiki/Python_(langage)) et publié sous [licence MIT](https://fr.wikipedia.org/wiki/MIT_Licence).

SQLAlchemy a opté pour l'utilisation du pattern Data Mapper plutôt que l'[active record](https://fr.wikipedia.org/wiki/Active_record_(patron_de_conception)) utilisés par de nombreux autres ORM.

SQLAlchemy a été publié en [février](https://fr.wikipedia.org/wiki/F%C3%A9vrier_2006) [2006](https://fr.wikipedia.org/wiki/2006_en_informatique) et est rapidement devenu l'un des ORM les plus utilisés par la communauté Python.

**Structure de SQLAlchemy :**

****

**Pickle :**

Un pickle est un objet Python représentant une chaine d'octets. Cela a l'air parfaitement simple ? Eh bien, c'est aussi simple ! Ce processus est appelé Pickling. Donc nous avons converti avec succès notre objet en une chaîne d'octets, maintenant comment pouvons-nous revenir en arrière ? Dépickler signifie reconstruire l'objet Python depuis la chaine picklée d'octets. Pour parler strictement, ce n'est pas une reconstruction dans un sens physique—cela signifie simplement que si nous avons pickler une liste, L, alors après dépicklage, nous retrouvons les contenus de la liste simplement en accédant à L.

Les termes « pickler » et « depickler » se rapportent respectivement à la sérialisation et à la désérialisation objet, qui sont des termes neutres par rapport au langage relatif à un processus qui transforme arbitrairement des objets complexes en représentations textuelles ou binaires de ces objets et inversement.

**ArgParser :**

Le module argparse est un "Command Line Parsing Module". En bon français, on dirait que le module argparse permet de parser correctement les arguments de la ligne de commande. Le module argparse est le successeur du module **optparse**. Ce module est inclus en standard dans python 2.7 et python 3.2. Vous pouvez sinon l'installer par un pip install argparse.

# Test

# Conclusion

En définitive, ce projet nous a appris à programmer en python et nous a apporté quelques connaissances dans l’utilisation des modules python déjà prêts. Il a également été un bon entraînement dans la programmation événementielle. Au début, nous avons beaucoup réfléchie à la problématique de l'analyseur de modification redondante qui était notre première idée de projet mais, comme nous l’avons expliqué dans l'introduction, nous nous sommes plutôt dirigés vers de réutilisation des modules comme pickle, SQLAlchemy, argparse. Peut-être que nous le développerons un jour si nous en avons le temps.

A partir de cette base de script, on peut facilement et rapidement créer tous les outils que le veut, c'est l'un des avantages de python.