

Programmation Objet

1-0: Présentation du cours et des outils

VCOD 2022/2023

Maxime Raynal

Plan du jour

- Présentation du cours
- Le langage python
- Quelques outils pratiques
- Gentle Python reminders (si en salle machine)

Présentation du cours

Le cours

- **Objectifs:**

- Introduction aux paradigme de la programmation orientée objet
- Algorithmes basés sur les objets

- **Langage utilisé:**



- **Outils annexes:**



Le cours

- L'enseignant: Maxime Raynal.
 - Contact: maxime.raynal@univ-grenoble-alpes.fr
- Planning: les lundis matin de 8h à 10h (dix séances)
- Examens et contrôle continu:
 - Trois composantes: quicks (mini-contrôles), TPs/projet, examen final.

Le cours

- Si vous ne comprenez pas, demandez !
- Plagiat !
- Pas mal de travail et de concepts à retenir, restez à jour.
 - Il faut consulter les ressources externes (beaucoup sont en anglais)
 - Travail à la maison
 - Concepts de code fondamentaux
- Ressources du cours:
 - Slides sur ma [page perso](#)
 - Notebooks et code sur le repo du cours: **TBD**

Le langage python™

Le langage python™

- Créé par Guido van Rossum (BDFL)



Le langage python™

- Créé par Guido van Rossum (BDFL)
- Le nom Python vient des Monty Python



Le langage python™

- Créé par Guido van Rossum (BDFL)
- Le nom Python vient des Monty Python
- Un des langages de programmation les plus utilisés au monde

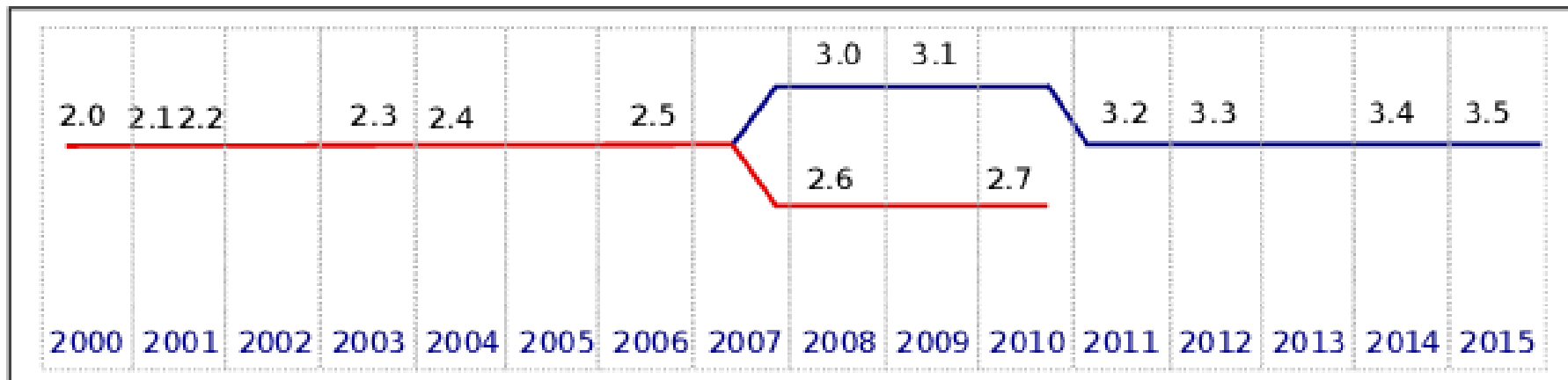


Le langage python™

- **Les versions**

- Débuts en 1989
- Python 0.9 en 1991 ; Python 1.0 en 1994 ; Python 2.0 en 2000
- Python 3.0 en 2008 ; 3.10 actuellement

- **Attention: Python 2 et 3 non-compatibles !!!!**



Le langage python™

- Un langage **interprété** (script)
- Un langage **multi-paradigme**:
 - Impératif
 - Fonctionnel
 - **Programmation orientée objet**

Le langage python™

- **Points forts:**

- **Versatilité**
- **Gigantesque écosystème** de bibliothèques
- **Facilité** de rédaction et de lecture du code
- Multi-plateformes

- **Points faibles:**

- Langage interprété
- Lenteur pour certaines tâches
- Gourmand en mémoire
- Peu utilisé sur mobiles

Le langage python™

- **Usages principaux:**

- Calcul scientifique, statistiques
- Machine learning (ML) et IA
- Data mining
- Prototypes
- Web
- Automatisation de tâches

- **Mais aussi**

- Jeux vidéos
- Langage à tout faire
- Etc ...

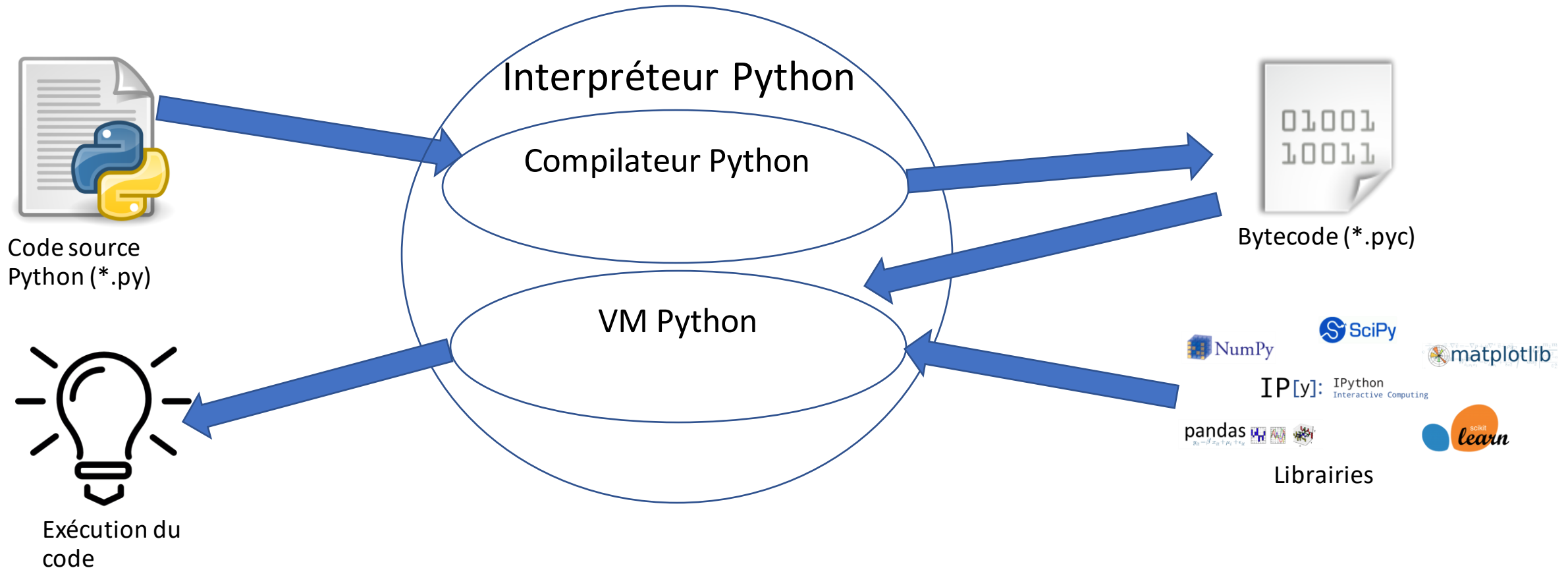
Le langage python™

À savoir:

- Il faut respecter les conventions d'écriture: [PEP-8](#)
- Suite de test intégrée: [pytest](#)
- Fonctionne avec un **interpréteur Python**
 - Responsable d'exécuter le code, qu'il soit dans une console, un fichier ou un notebook.
 - La version de Python est celle de l'interpréteur
- Les librairies tierces sont installées dans un environnement Python
 - Windows: utilisation de [Anaconda](#) vivement conseillée
 - Linux & Mac: [pip](#) ou [Anaconda](#)
- La documentation standard ([docs.python.org](#)) est très pédagogique, complète, juste et à jour.

Le langage python™

- Un langage **interprété** (script)




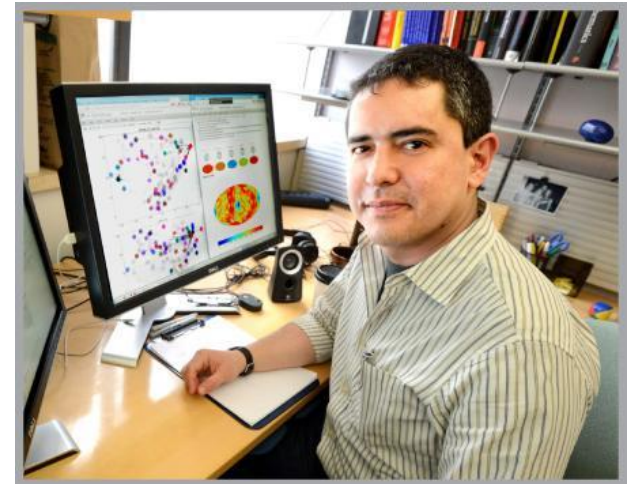
Le langage python™

- Rappels sur le langage pendant la 2e heure.
- Série d'exercices pour vous remettre dans le bain pour la semaine prochaine.

Les notebooks



- Nom dérivé de trois langages de programmation: Julia, Python, R.
- Spin-off de son prédécesseur iPython en 2014 ; créé par Fernando Pérez
- Environnement de programmation langage-agnostique et interactif destiné à produire des documents.
- Fonctionne dans un navigateur web.
- Interprète nativement le **markdown**  et **Python**.



Les notebooks



- Usages:
 - Créer des modèles statistiques
 - Entraîner des modèles de ML et d'IA
 - Visualiser des données
 - Support de cours interactif
 - Mais pas que !
- Outil très pratique et populaire
- Ne fonctionne pas uniquement avec Python mais une centaine de langages!

Les notebooks



- Usages:
 - Créer des modèles statistiques
 - Entraîner des modèles de ML et d'IA
 - Visualiser des données
 - Support de cours interactif
 - Mais pas que !
- Outil très pratique et populaire
- Ne fonctionne pas uniquement avec Python mais une centaine de langages!



- Logiciel de versions décentralisé
- Écrit par Linus Torvald en 2005
- Outil **indispensable** dans la panoplie d'un développeur.
- Les notebooks seront stockés sur un dépôt (repository ou repo) git





- Logiciel de versions décentralisé
- The stupid content tracker

```
Linus Torvalds Initial revision of "git", the information manager from hell

1 contributor

168 lines (135 sloc) | 8.2 KB

1
2     GIT - the stupid content tracker
3
4     "git" can mean anything, depending on your mood.
5
6     - random three-letter combination that is pronounceable, and not
7       actually used by any common UNIX command. The fact that it is a
8       mispronunciation of "get" may or may not be relevant.
9     - stupid. contemptible and despicable. simple. Take your pick from the
10      dictionary of slang.
11    - "global information tracker": you're in a good mood, and it actually
12      works for you. Angels sing, and a light suddenly fills the room.
13    - "goddamn idiotic truckload of sh*t": when it breaks
14
15    This is a stupid (but extremely fast) directory content manager. It
16    doesn't do a whole lot, but what it does do is track directory
17    contents efficiently.
18
```

Mini-projet

- La semaine prochaine (ou la suivante) je vous donnerai des sujets au choix pour un mini-projet
 - Fait partie de la note finale.
 - Projet de groupe (idéal -> 3 personnes).
 - Orienté statistiques et apprentissage machine.
 - Idées de sujet: inférence de langue, génération de texte, génération de cartes à partir de données GTFS, fractales de Newton
- Pour la semaine prochaine: formez des groupes de trois.

Coding time

- Objectifs:
 - Découvrir les notebooks jupyter.
 - Premiers pas en markdown.
 - Gentle Python reminders.

- Marche à suivre:
 - Télécharger le notebook sur **TBD**
 - Lancez jupyter
 - Ouvrez le notebook puis suivez les instructions