

# Infrastructure informatique du DI

## Présentation

Gabriel Girard

Département d'informatique



5 janvier 2018

# Infrastructure informatique du DI

- 1 Infrastructure informatique du DI
  - Laboratoires
- 2 Environnements logiciels
  - Environnement Windows
  - Environnement Linux
  - Logiciels gratuits
  - Information utile
- 3 Utilisation de PyCharm
  - Démarrage de PyCharm
  - Création d'un nouveau programme
  - Ouverture d'un programme existant
- 4 Compilation d'un programme

# Laboratoires d'enseignement

- Lab. Windows/Linux  $\Rightarrow$  D4-1017 (Windows/Linux)
- Lab. Windows/Linux  $\Rightarrow$  D4-1023 (Windows/Linux)
- Lab. d'acquisition  $\Rightarrow$  D4-023 (Windows/Linux)
- Lab. Windows/Linux  $\Rightarrow$  D4-022 et 024 (Windows/Linux)
- Salle de travail (Le RIRE<sup>1</sup>)  $\Rightarrow$  D4-1021
- Serveurs de fichiers réseau

---

1. Refuge Informatique pour la Réflexion et l'Échange

# Laboratoires d'expérimentation logicielle (Windows/Linux)

Local et nombre de postes				
D4-1017	D4-1023	D4-023	D4-024	D4-022
23	28	10	10	4

- Linux (Ubuntu 16.04) ou Windows 10
- Code d'accès requis (CIP)
- Imprimante au laser
- Accès à des dossiers réseaux

# Impression laser

- Accessible de Windows ou Linux
- Nom de l'imprimante = `DInf-numéro_du_local-Laser`  
Exemple pour le local D4-1017 : `DInf-D4-1017-Laser`
- Vous devez acheter des impressions  
Pour informations : <http://usherbrooke.ca/simus>

# Lecteur réseau privé

## ① Pour les postes des laboratoires

- Windows : U
- Linux : Personnel

## ② Pour un poste personnel

- Monter comme lecteur réseau via VPN
- Windows : smb, sftp, FileZilla
- Linux : smb, Nautilus, sshfs, FileZilla, sftp
- Pour informations :

**<http://www.usherbrooke.ca/informatique/>**

→ **Intranet**

→ **Ressources et documentation**

→ **F.A.Q.**

→ **Accès aux lecteurs réseaux**

## ③ N.B. : ne rien laisser localement

# Lecteur réseau public

- 1 Pour les postes des laboratoires
  - Windows : P
  - Linux : Public
- 2 Pour un poste personnel

**<http://www.usherbrooke.ca/informatique/>**

→ **Intranet**

→ **Ressources et documentation**

→ **F.A.Q.**

→ **Accès aux lecteurs réseaux**

# Environnement Windows

- Windows 10 et son GUI
- Mode graphique ou mode ligne (Dos)
- Langages : Python2, Python 3 ...
- Éditeurs : Wordpad , Notepad, NotePad++ ...
- Bureautique : Office, LibreOffice, ...
- IDE : PyCharm, Spyder3, ipython, Jupyter, ...
- Accès à GitLab
- Liste complète disponible sur le WEB
  - <http://www.usherbrooke.ca/informatique/>
  - **Intranet**
  - **Ressources et documentation**
  - **Nos laboratoires**



# Environnement Linux

- Linux Ubuntu 16.04 (Unix)
- GUI : Unity (Gnome, Xfce, Kde, Wmaker, ...)
- Mode graphique ou mode ligne (xterm, gterm, ...)
- Langage : Python 2, Python 3 ...
- Bureautique : LibreOffice, latex, dia, xfig, ...
- Éditeurs : nedit, gedit, ...
- IDE : Pycharm, Spyder3, ipython, Jupyter, ...
- Accès à GitLab
- Liste complète disponible dans le menu et dans l'intranet

# Logiciels gratuits

## Environnement Windows

- Logiciels Microsoft gratuits aux étudiants (MSDNaa)
- Cela inclut : systèmes et outils de développement
- Consultez :

**<http://www.usherbrooke.ca/informatique/>**

→ **Intranet**

→ **Ressources et documentation**

→ **Logiciels - services - outils**

→ **Logiciels Microsoft gratuits**



# Logiciels gratuits

## Environnement Windows et Linux

- Logiciels VMWare gratuits aux étudiants (ELMS)
- Consultez :
  - <http://www.usherbrooke.ca/informatique/>
  - **Intranet**
  - **Ressources et documentation**
  - **Logiciels - services - outils**
  - **Logiciels VMware gratuits**

# Logiciels gratuits

## Machines virtuelles

- Diverses machines virtuelles sont disponibles
  - «*Lecteur réseau public*»/logiciels
- Deux machines virtuelles Linux
  - 1 Copie de notre environnement
  - 2 Minimale
- Une machine virtuelle Windows
  - Obtention d'une licence requise

# Environnement Unix

## Mode ligne

- Valide pour Linux et Solaris
- Fenêtre : gterm (xterm, ...)
- Shell : sh, csh, bash, tcsh, zsh, ...
- Commandes Unix et applications

<http://www.usherbrooke.ca/informatique/>

→ Intranet

→ Ressources et documentation

→ Logiciels - services - outils

→ Introduction à Unix - Solaris

# Environnement Unix

## Tcsh ou Bash

- Commandes de lignes Unix
- ↑ : commande précédente
- ↓ : commande suivante
- !aa : rappel cmd débutant par "aa"
- !xxx : cmd # xxx de l'historique (history)
- tabulateur : complète le nom
- Ctrl-D : liste de correspondances (ou logout)
- ^ xx^ yy : rappel dernière cmd modifiée

# Environnement Unix

## Commandes de base

- passwd
- man
- who, w, users, rwho, rusers
- quota -v
- uname -r
- ps -u *nom*, kill -9 #
- logout (Ctrl-d)
- vi, vim

# Environnement Unix

## Commandes de base (suite)

- mkdir, rmdir, rm - r (-r )
- cd *nom*, cd .., cd
- more , less
- cp, mv, rm
- pwd
- l, dir, ls -la
- lp -d *nom*, lpstat, cancel *no*



# Environnements

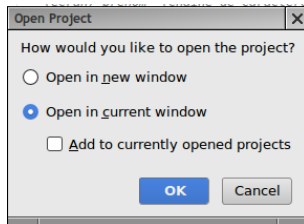
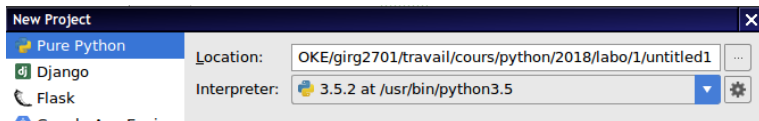
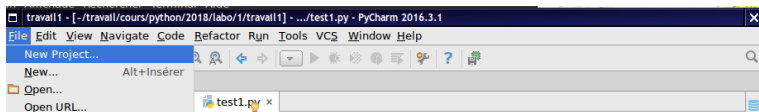
## Autres considérations

- Mettre vos programmes sur votre Lecteur réseau personnel  
**N.B. : ne rien laisser localement**
- Passage Windows → Unix : dos2unix
- Les tabulateurs : configuration ou expand

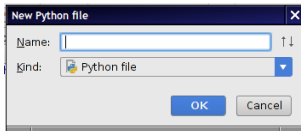
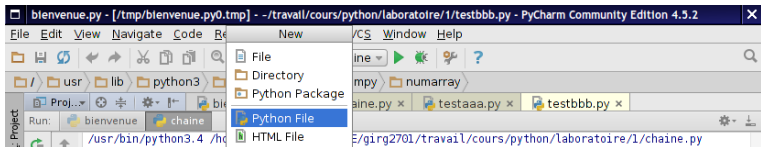
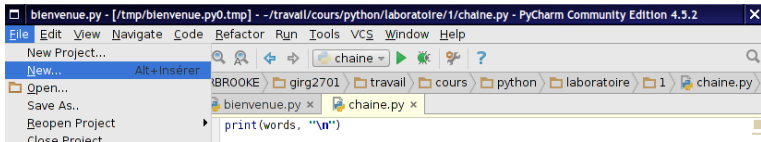
# Démarrer PyCharm

- 1 Connexion sur Windows ou Linux
- 2 Démarrer PyCharm
  - Démarrer l'application à partir du menu
  - «Cliquer» sur l'application avec le bouton de droite et choisir PyCharm

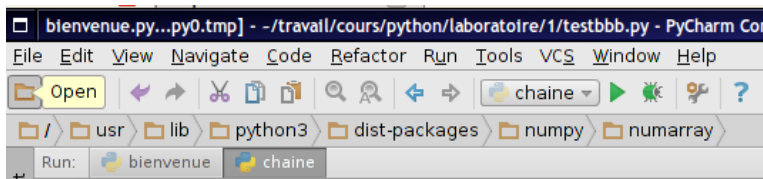
# Créer un projet dans PyCharm



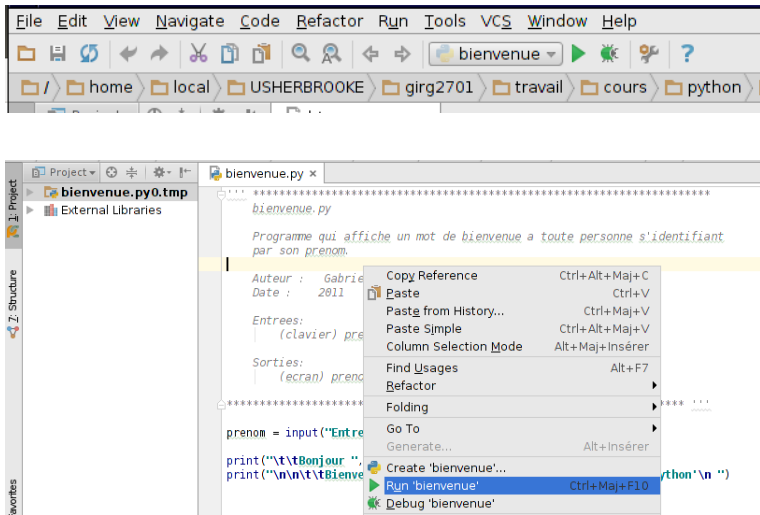
# Créer un fichier dans PyCharm



# Ouvrir un fichier existant dans PyCharm



# Exécuter avec PyCharm



# Exécuter avec PyCharm

- 1 Télécharger les programmes à partir du site Web du cours (Laboratoire 1)
- 2 Ouvrir le fichier avec PyCharm
- 3 Exécuter les programmes

# Compiler et exécuter en ligne de commande

① Ouvrir un terminal

② Exécution

```
python3 nom_fichier.py
```

③ Exécution et redirection des E/S (Linux)

- Sortie à l'écran

```
python3 nom_fichier.py
```

- Lecture dans un fichier

```
python3 nom_fichier < nom_fichier_entree
```

- Sortie dans un fichier

```
python3 nom_fichier.py > nom_fichier_sortie
```

- Lecture et sortie dans des fichiers

```
python3 nom_fichier.py < nom_fichier_entree >  
nom_fichier_sortie
```



# Exemple

## ① Exécution (Linux)

### ① Sortie à l'écran

```
> python3 test1.py
```

### ② Entrée à partir d'un fichier

```
> python3 test1.py < entree
```

### ③ Sortie dans un fichier

```
> python3 test1.py > resultat
```

### ④ Les deux

```
> python3 test1.py < entree > resultat
```