Tutoriel PyCharm

Kévin Huguenin et Timothy Bollé 20/02/2017

Installation

Si vous utilisez les machines de la salle informatique, les programmes étant déjà installés, vous pouvez directement passer à l'étape d'utilisation (étape 13).

Installation de Python

Pour utiliser PyCharm, il est nécessaire que Python soit installé sur votre ordinateur. Si ce n'est pas le cas, Python peut être téléchargé à l'adresse suivante : <u>https://www.python.org/downloads/</u>. Assurez-vous de bien noter dans quel dossier vous l'avez installé.

Installation de PyCharm

PyCharm est un IDE (*Integrated Development Environment*). Il s'agit, tout comme <u>jEdit</u>, d'un logiciel permettant d'intégrer dans une même fenêtre tous les éléments utiles à la programmation en python : un éditeur de texte pour écrire des scripts, une console pour exécuter des programmes, ainsi qu'un explorateur de fichiers pour parcourir le projet en cours.

En tant qu'étudiant, vous pouvez utiliser la version professionnelle de PyCharm gratuitement. Pour cela rendez-vous sur la page <u>https://www.jetbrains.com/student/</u> et appuyez sur le bouton **Apply Now**. Vous serez amenés sur une page vous demandant votre nom et prénom, ainsi que votre adresse email académique. Il est ici important d'entrer votre **adresse UNIL**. Vous recevrez ensuite un email avec des instructions sur la façon d'activer votre licence. Suivez-les. Il vous sera demandé de créer un compte JetBrain (la société produisant PyCharm). Il vous suffit de créer un identifiant (vous pouvez utiliser votre adresse email de l'Unil comme identifiant facile à retenir), ainsi qu'un mot de passe. Une fois connecté à votre compte, vous pourrez télécharger la version Professionnelle de Pycharm.

Vous pouvez maintenant exécuter l'installeur de <u>PyCharm</u> que vous venez de télécharger en double cliquant dessus. Pour les utilisateurs de Mac, il se peut que vous deviez autoriser l'installation d'applications externes. Pour cela, allez dans l'application *Préférences Système*, puis dans l'onglet *Sécurité et confidentialité*. Ici vous pouvez ouvrir le cadenas et cocher l'option *N'importe où*.

1. Cliquez sur Next.

🗿 PyCharm Setup	- 🗆 X
PC	Welcome to the PyCharm Setup Wizard
	This wizard will guide you through the installation of PyCharm.
	It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.
	Click Next to continue.
	Next > Cancel

2. Ici vous pouvez choisir le dossier où vous voulez installer PyCharm. Next.

-	Chasses Inst	all Location			
PC	Choose inst	an Location			
	Choose the fo	older in which to in	stall PyCharm.		
Setup will install PyCha Browse and select ano	arm in the following ther folder. Click Ne	folder. To install in xt to continue.	a different fold	der, click	
Doctination Folder					
Destination Folder	arm 2016 3 2		B	'NWSP	7
Destination Folder E:\PyCharm\PyCha	arm 2016.3.2		Br	'owse]
Destination Folder E:\PyCharm\PyCha Space required: 580.0	arm 2016.3.2 MB		Br	owse]
Destination Folder E:\PyCharm\PyChz Space required: 580.0 Space available: 708.5	arm 2016.3.2 MB IGB		BI	owse]

3. Ici vous pouvez cocher la case <u>Create</u> associations .<u>py</u>. Ensuite, faites Next.

PC	Installation Opt	ions		
	Configure your F	yCharm installatio	n	
Create Desktop shortcu	ut			
32-bit launcher				
Create associations				
			1	

4. Cliquez ensuite sur Install puis sur Finish, une fois l'installation terminée.

Utilisation

Configuration initiale

Pour cette première utilisation, nous allons configurer PyCharm. Une documentation complète de PyCharm est disponible à l'adresse https://www.jetbrains.com/pycharm/documentation/.

- 5. Démarrez PyCharm
- 6. Sélectionnez I do not have... puis appuyez sur OK



7. Acceptez les conditions d'utilisation

	laiuons:
JetBrains Privacy	Policy
Last updated: 14th March 2016	
This Policy may be amended from time to tir the point of time of the purchase/registrati (whichever occurs later) shall apply. The da hřebenech II 1718/10, PSČ 147 00, Česká	me. The respective latest version of the policy at on of a JetBrains Software Product ata controller is JetBrains s.r.o., Praha 4, Na republika.
In this Privacy Policy, we describe the type "data"), that we collect from you when you webSite) and certain JetBrains products ar (collectively, our "services") and how we us definitions will be used throughout this Priv.	of data, including personal data (collectively, use our Website (listed under JetBrains di services as described in this Privacy Policy se and disclose that data. The following acy Policy.
Definitions	
 <u>Definitions</u> <u>Why We Collect Data</u> 	

8. Si vous utilisez la version professionnelle, entrez vos identifiants puis cliquez sur Activate.

PyCharm License Activation	- 🗆 ×
Activate C Evaluate for free	Buy PyCharm
Activate license with:	de 🔿 License server
Username or email:	
Password:	Forgot?
	Activate Exit

Sur la fenêtre suivante, vous pourrez choisir les thèmes d'affichage des fenêtres et de l'éditeur. Ce sont des réglages esthétiques à effectuer selon vos préférences. Vous pouvez regarder les différents thèmes proposés pour l'éditeur (*Editor colors and fonts*) et voir à quoi cela ressemble en cliquant sur le bouton *Click to preview*.

9. Cliquez sur OK.



10. Cliquez sur **Configure > Settings**.

Welcome to Pychami		-	
	PC		
	PyCharm Version 2016.3.2		
	* Create New Project		
	🦮 Open		
	Check out from Version Control +		
	Country of	Continue	Cat Halp
	1 Events +	Configure -	Get Help
	€ Events →	Configure Settings Plugins	Get Help
	Events •	Configure Settings Plugins Import Settings	Get Help
	Events •	Configure Settings Plugins Import Settings Export Settings	Get Help
	1 Events 🗸 🕚	Export Settings Plugins Import Settings Export Settings Settings Reposi	Get Help

11. Dans la nouvelle fenêtre, allez dans l'onglet **Project Interpreter** puis cliquez sur la petite flèche en haut à droite. Dans le menu déroulant, il devrait y avoir un chemin vers votre exécutable pyhton python.exe. Sélectionnez le. Il est normal de ne pas observer la même liste dans l'exemple de tableau ci-dessous.

	/ Project interpreter @ Por deradic proje	a		
Appearance & Behavior	Project Interpreter: 👘 3.6.0 (E:\Anacon	da\envs\python36\python.exe)		- 4
Keymap				
Editor	Package	Version	Latest	
Plugins	Babel	2.3.4	2.3.4	
ringins	Bottleneck	1.2.0	1.2.0	
Version Control	Cython	0.25.2	0.25.2	
Project Interpreter	Flask	0.12	0.12	
Build Execution Deployment	Flask-Cors	3.0.2	3.0.2	
build, execution, Deployment	HeapDict	1.0.0	1.0.0	
Languages & Frameworks	Jinja2	2.9.4	➡ 2.9.5	
Tools	MarkupSafe	0.23	0.23	
	Pillow	4.0.0	4.0.0	
	PyYAML	3.12	3.12	
	Pygments	2.1.3	➡ 2.2.0	
	QtAwesome	0.4.3	➡ 0.4.4	
	QtPy	1.2.1	1.2.1	
	SQLAIchemy	1.1.5	1.1.5	
	Werkzeug	0.11.15	0.11.15	
	XIsxWriter	0.9.6	0.9.6	
	_license	1.1		
	alabaster	0.7.9	0.7.9	
	anaconda	4.3.0		
	anaconda-client	1.6.0	1.2.2	
	anaconda-navigator	1.4.3		
	astroid	1.4.9	1.4.9	
	astropy	1.3	1.3	
	babel	2.3.4		
	backports	1.0	1.0	
	backports.shutil-get-terminal-size	1.0.0		
	beautifulsoup4	4.5.3	4.5.3	

12. Allez ensuite dans l'onglet **Build, Execution, Deployment > Console > Python Console**. Sélectionnez à nouveau le chemin vers votre exécutable python à l'aide de la flèche dans la ligne **Python interpreter**. Sélectionnez ensuite le dossier où vous souhaiterez faire vos exercices dans la ligne Working directory, à l'aide du bouton sur la droite. Nous vous conseillons de **faire vos exercices dans un sous dossier de votre répertoire SwitchDrive**. Cela vous permettra d'y accèder depuis n'importe quel ordinateur ayant un accès à votre SwitchDrive. Cliquez ensuite sur **OK**. Une barre de chargement apparaît. Laissez la finir.

C Default Settings		×
Q. → Appearance & Behavior	Build, Execution, Deployment > Console > Python Console For default project Tenvironment	
Keymap ▶ Editor	Project: [none]	
Plugins Version Control	Bython interpreter: 👘 Python 3.6.0 (E\Anaconda\envx\python36\python.exe)	
Build, Execution, Deployment Debugger	Working directory: E\Switch_Drive\Informatique_2\Exercices	
Python Debugger Deployment Buildout Support Console	Add content roots to PYTHONPATH Add source roots to PYTHONPATH Starting script	
Python Console Coverage Docker Docker Registry Language & Frameworks Tools	import sys print("Python %s on %s % (sys.version: sys.platform)) sys.path.extend[[WORKING_DIR_AND_PYTHON_PATHS]]	
	OK Cancel Apply	Help

quelques

opérations nous ont permis d'indiquer à PyCharm quel Python nous souhaitons utiliser et où nous souhaitons travailler.

Travailler sur un projet

Nous allons commencer par créer un nouveau projet.

13. Appuyez sur Create New Project.

Welcome to PyCharm		<u></u>		×
	PC			
	PyCharm Version 2016.3.2			
	★ Create New Project ► Open			
	Check out from Version Control -			
	Events • * Cor	ifigure 👻	Get Help) *

14. Sélectionnez ensuite l'onglet Pure Python et assurez-vous que le dossier de travail est celui désiré (sous dossier SwitchDrive par exemple). Vous pouvez donner un nom au dossier du projet. Appuyez ensuite sur **Create**.

Rew Project			<u></u>		×
Pure Python	Location:	E:\Switch_Drive\Informatique_2\Exercices\Demo			
 Django Flask Google App Engine Pyramid Web2Py Angular CLI Angular/S Foundation HTML5 Boilerplate React App React Native Twitter Bootstrap Web Starter Kit 	Location: Interpreter:	E:\Switch_Unive\Informatique_Z\Exercices\Demo			*
¢				Crea	ate

15. Une fois arrivé sur l'écran principal de PyCharm, vous pouvez créer votre premier script python. Pour cela, en haut à gauche, cliquez sur **File > New... > Python File**.

3 Open Open URL Save As Open Becent Close Project Import Settings Pefault Settings Eport Settings Export Settings Eynort Settings Synchronize Ctrl+Alt+V Import Settings Synchronize Ctrl+Alt+V Import Settings Pint Add to Favorites Pint Add to Favorites Pint Add to Favorites Pint Exectory Read-only Power Save Mode Egit	New		Alt+Insérer	⊕ ÷	- I-				
Open DRL Save As Open Becent * Getings Close Project * Settings Default Settings Import Settings Epont Settings Settings Settings Sync All Chrl- Alt-Y Invalidate Caches / Restart Pint Add to Favorites File Encoding Line Separators Make Directory Read-only Power Save Mode Egit	Open.			matique_2\Exercic	es\Demo		New		
Save As. Open Becent Settings Default Settings Epont Settings Epont Settings Settings Repository Seve All Chri+S Synchronize Chri+Alt+Y Invalidate Caches / Restart Pint Add to Favorites File Encoding Line Separators File Encoding File Encoding F	Open	JRL				📋 File			
Open gecent Python Package Close Project Python Package * Settings Ctrl+Alt+S Default Settings Python Settings Sport Settings Strings Repository Setting Repository Ctrl+S Synchronize Ctrl+Alt+Y Invaluate Caches / Restart Data Source Pint Add to Fayorites File Encoding Line Separators Make Directory Read-only Pawer Save Mode Egit Egit	Save A	5				Dire	ctory		
Close Project Settings Settings Ctrl+Alt+S Defgult Settings Stylesheet import Settings CoffeeScript File Synchronize Ctrl+Alt+V Invalidate Caches / Restart CoffeeScript File Print Data Source Add to Fgvorites File Encoding Line Separators Mare: Power Save Mode Imme: Egit Egit	Open j	Secent				Pyt	non Package		
 Settings Ctrl+Alt+S Defgult Settings Export Settings Export Settings Export Settings Systemet Stylesheet Systemet 	Close	roject	200 2002			ile Pyti	non File		
Default Settings H1ML File Import Settings Export Settings Sport Settings JavaScript File Settings Repository Coffeescript File Swe All Ctrl+S Synchronize Ctrl+Alt+Y Invalidate Caches / Restart Data Source Print Add to Favorites Add to Favorites File Encoding Line Separators Mare: Power Save Mode Egit	P Setting	5	Ctrl+Alt+S		_	Jup:	ter Notebook		
Import Settings Export Settings Settings Repository Settings Repository Synchronize Ctrl+ Alt+Y Invalidate Caches / Restart Pint Add to Favorites File Encoding Line Separators Make Directory Read-only Power Save Mode Egit	Defaul	t Settings				HII Chul	AL File		
Export Settings Settings Repository § ave All Ctrl+S Synchronize Synchronize Invalidate Caches / Restart Print Add to Favorites File Encoding Line Separators Make Directory Read-only Power Save Mode Egit	Import	Settings				and Java	Script File		
Settings Repository § gave All Ctrl+S § synchronize Ctrl+Alt+V Invalidate Caches / Restart Print Add to Fgvorites File Encoding Line Separators Make Directory Read-only Power Save Mode Egit	Export	Settings				Tvp	eScript File		
l gave All Ctrl+S Synchronize Ctrl+Alt+Y Invalidate Caches / Restart P print Add to Favorites File Encoding Line Separators Make Directory Read-only Power Save Mode Egit	Setting	is Repository	·			Cof	feeScript File		
Synchronize Ctrl+Alt+Y Invalidate Caches / Restart	E Save A	11	Ctrl+S			🔛 Ghe	rkin feature file		
Invalidate Caches / Restart Print Add to Favorites File Encoding Line Separators Make Directory Read-only Power Save Mode Egit	Synchi	onize	Ctrl+Alt+Y			Edit	File Templates		
Pint Add to Favrites File Encoding Line Separators Make Directory Read-only Power Save Mode Egit	Invalid	ate Caches /	Restart			E Dat	Source		
Add to Favorites File Encoding Line Separators Make Directory Read-only Power Save Mode Egit	Print								
File Encoding Image: New Python file X Line Separators Make Directory Read-only 11 Power Save Mode Kind: Python file Egit Image: New Python file Image: New Python file	Add to	Favorites	•						
Line Separators Make Directory Read-only File Egit	File En	coding				PC New	Python file		×
Make Directory Read-only Power Save Mode Ejit Ejit	Line Se	parators	•						
Power Save Mode Kind: Kind:	Make	Directory Rea	ad-only			Name:	Demoj		
Exit	Power	Save Mode				Kind:	🛃 Python file		•
	Exit								
OK Cancel								ок	Cancel

16. Le fichier sera automatiquement placé dans le dossier de votre projet.

Demo) 🔒 Demo.py) Ex	plorateur de fichie	rs	
Project *	〇 本 奋· It-	Demo.pv ×	
Demo E:\Switch Drive\Inform	atique 2\Exercices\Demo	1 Deniopy	
Demo.pv		2 Ceci	est un programme de demo!
Illi External Libraries		3	
		4 0	
		5	
		7 prin	c(phrase)
		8	
Explorateur de p	roiet	9 for	i in range(len(phrase)):
Explorateur de p	rojet	10	<pre>print(i*" "+phrase[i])</pre>
			Editeur de texte
Accès au	consoles		
/ nuces du	CONSOLCS		

17. Pour ouvrir un projet déjà créé, allez dans l'onglet File > Open. De là, vous pouvez ouvrir un projet complet ou juste un script. Les dossiers contenants des projets apparaissent avec un carré noir au coin du dossier. Pour l'ouvrir, sélectionnez le dossier ou le fichier voulu et faites Ok.



18. Pour exécuter votre script, allez dans l'onglet **Run > Run¹**, puis cliquez sur le fichier à exécuter.

Demo 👌 🐻 Demo.py 🔪	Run	Maj+F1
Project 🔻	🗰 <u>D</u> ebug	Maj+F
Demo E:\Switch Drive\Informatique	🕅 Run with Co <u>v</u> erage	
Demo.pv	Profile	
Illi External Libraries	Example 2 Concurrency Diagram for	
	🕨 Run	Alt+Maj+F1
	👯 Debug	Alt+Maj+F
	Attach to Local Process	
	Edit Configurations	
	Import Test Results	
	Stop	Ctrl+F2
	📳 Show Running List	
	▼ Step <u>O</u> ver	F
	Force Step Over	Alt+Maj+F
	🔟 Step Into	F
	Market Force Step Into	Alt+Maj+F
	🖄 Smart Step Into	Maj+F
	💻 Step Ou <u>t</u>	Maj+F8
	[™] £ Run to <u>C</u> ursor	Alt+F
	[™]): Force Run to Cur <u>s</u> or	Ctrl+Alt+F
	Resume Program	F
	Evaluate Expression	Alt+F
	Quick Evaluate Expression	Ctrl+Alt+F
	E Show Execution Point	Alt+F1
	Toggle Line Breakpoint	Ctrl+F
	Toggle Temporary Line Breakpoint	Ctrl+Alt+Maj+F
	Toggle <u>B</u> reakpoint Enabled	
	Siew Breakpoints	Ctrl+Maj+F

Une nouvelle zone apparaît. Il s'agit de la console. C'est ici que vous observerez les résultats de vos scripts.

19. Par défaut, l'output d'un script est renvoyé dans la console Run.

¹ Vous pouvez configurer les raccourcis clavier en allant dans les paramètres de Pycharm (**File > Settings**) puis dans l'onglet **Keymap**.

Ru	in 🟓 De	emo
	• + • +	E:\Anaconda\envs\python36\python.exe E:/Switch_Drive/Informatique_2/Exercices/Demo/Demo.py Hello world! H
П	<u>.</u>	e
		1
		0
×		W
-		0
1		r
		1
		1
		Process finished with exit code 0
	4: Run	🔮 📴 TODO 🛛 🍓 Python Console 🛛 Terminal
1		

20. Vous pouvez aussi utiliser la console python pour effectuer des tests indépendamment de ce que vous mettez dans vos scripts.



Comme vous pouvez le voir, même après l'exécution du script, les variables du script n'existent pas dans la console python. Il est toutefois possible d'exécuter une partie de votre script dans la console. Pour cela, sélectionnez les lignes à exécuter, faites un clic droit et appuyez sur **Execute Selection in Console**. Faire cela produit le même effet que si vous aviez tapé toutes les instructions sélectionnées dans la console.



21. Vous avez aussi accès à un terminal de commande de votre système d'exploitation.



Redirection de l'input et de l'output standard

22. Nous avons vu que normalement, le résultat d'un script (son output) est renvoyé dans la console Run. Il vous est aussi possible de rediriger l'output dans un fichier. Pour cela, allez dans l'onglet Run > Edit Configurations.... Allez ensuite dans l'onglet Logs et cochez la case Save console output to file en choisissant le nom du fichier dans lequel vous voulez enregistrer l'output.

Run/Debug Configurations		×
+ - □ H 9 + + = 13 ▼	Name Demo Share	Single instance only
▶ 9 ^c Defaults	Log files to be shown in console	
	Is Active Log File Entry	Skip Content +
	No log files attached	-
	Save console output to file: E:\Switch_Drive\Informatique_2\Exercices\Demo\output.txt	
	Show console when a message is printed to standard output stream Show console when a message is printed to standard error stream	
	* Refore Jaunch: Activate tool window	
	+ - / ↑ ↓	
	There are no tasks to run before launch	
	Show this page 🛛 Activate tool window	
	OK Cancel	Apply Help

23. Il vous est aussi possible de faire ces redirections en utilisant le terminal de commande.



Cette commande permet de dire à la machine d'exécuter *Demo.py* avec le programme *Python*, de transmettre en input standard le fichier *input.txt* (< **input.txt**), de rediriger l'output sur le fichier *output.txt* (> **output.txt**) et enfin de rediriger les erreurs dans le fichier *error.txt* (2> **error.txt**).

Installer des modules

PyCharm permet d'installer et de gérer des modules python à l'aide du module pip². Pour cela, ce dernier doit être installé sur la machine.

- 24. Dans PyCharm allez dans l'onglet File > Settings.
- 25. Allez ensuite dans l'onglet **Project > Project interpreter**. Ici s'affiche la liste des modules déjà installés. Appuyez sur le **plus** en haut à droite pour ajouter un module.

	Project: Projet_Demo > Project Interpro	eter 🐵 For current project		
Appearance & Behavior Keymap	Project Interpreter: 🟓 3.6.0 (E:\Anacor	nda\envs\python36\python.exe)		•
Editor	Package	Version	Latest	
Plugins	Babel	2.3.4	2.3.4	_
	Bottleneck	1.2.0	1.2.0	
Version Control	Cython	0.25.2	0.25.2	
Project: Projet_Demo	Flask	0.12	0.12	
	Flask-Cors	3.0.2	3.0.2	
Project Structure	HeapDict	1.0.0	1.0.0	
	Jinja2	2.9.4	➡ 2.9.5	
Build, Execution, Deployment	MarkupSafe	0.23	0.23	
Languages & Frameworks	Pillow	4.0.0	4.0.0	
Tools	PyYAML	3.12	3.12	
	Pygments	2.1.3	⇒ 2.2.0	
	QtAwesome	0.4.3	⇒ 0.4.4	
	QtPy	1.2.1	1.2.1	
	SQLAIchemy	1.1.5	1.1.5	
	Werkzeug	0.11.15	0.11.15	
	XIsxWriter	0.9.6	0.9.6	
	_license	1.1		
	alabaster	0.7.9	0.7.9	
	anaconda	4.3.0		
	anaconda-client	1.6.0	1.2.2	
	anaconda-navigator	1.4.3		
	astroid	1.4.9	1.4.9	
	astropy	1.3	1.3	
	babel	2.3.4		
backports 1.0	1.0			
	backports.shutil-get-terminal-size 1.0.0			
	han an hit han a h	450	450	

26. Dans cette fenêtre, vous pouvez rechercher un module et l'installer. Vous pouvez aussi spécifier la version du module ainsi que des options si nécessaire.

Q* beau		8
autiful-soup autofulsoup4	Description Screen-scraping library Version 45.3 Author Leonard Richardson matter leonardr@sedBaut.org http://www.crummu.com/software/ReautifulSoup/bs4/	
	Specify version 4.5.3	

Si vous travaillez sur un ordinateur de la salle informatique, vous devez indiquer l'option **–user** afin d'installer le module seulement pour l'utilisateur connecté.

² Une documentation des commandes disponibles avec pip est accessible à l'adresse <u>https://pip.pypa.io/en/stable/</u>

27. Vous pouvez aussi installer les modules en utilisant le terminal de l'ordinateur.



Trucs et astuces

28. **Auto-complétion** PyCharm propose un système d'auto-complétion qui permet de faciliter l'utilisation de nombreux modules.

Ceci est un programme de demo!	
- ·····	
import math	
phrase = "Hello world!"	
print(phrase)	
for a day warme (2 or (observable)	
for 1 in range (ien(phrase)):	
princ(r. +phrase(r))	
a = math.	
f acos (x)	math
() acosh (x)	math
(f) asin (x)	math
() asinh (x)	math
() atap (x)	math
() atan2 (v v)	math
() atanh (x)	math
() ceil(x)	math
(1) converge (x, y)	math
() cos (x)	math
	math
Press Ctrl+Point to choose the selected (or first) suggestion a	nd insert a dot afterwards >> 7

Lorsque vous écrivez, le logiciel vous propose les différentes possibilités selon ce que vous écrivez. Les différentes variables et fonctions contenues dans un module vous sont aussi proposées. Il vous est donc possible de choisir une des propositions et celle-ci sera écrite automatiquement.

29. Documentation Pycham propose également d'avoir accès à de la documentation concernant une variable ou une fonction. Pour cela, placez votre curseur sur un élément et allez dans l'onglet View > Quick Documentation³



30. **Définition de fonction** Vous pouvez accéder à la définition d'une fonction ou d'une variable. Pour cela, placez votre curseur sur un élément et allez dans **View > Quick Definition**.

³ Vous pouvez aussi utiliser les raccourcis clavier associés.

1	• ·····		
2	Ceci est un pro	gramme de demo!	
3			
1	- ·····		
	import math		
	phrase = "Hello	world!"	
	print (phrase)		
	for i in range((len(phrase)):	
	print(i*" "	+phrase[1])	
	a = math.cos(ma	th.pi)	
	a = math.cos(ma print(a)	Definition of pi	1
	a = math.cos(ma print(a)	Definition of pi	≯ 36∖python.exe) >
	a = math.cos(ma print(a)	Lth.pi) Definition of pi " math.py The C Python 3.6.0 (El-Anaconda)emv2/python pi = 3.141592653589793	∦ 36∖python.exe) >
	a = math.cos(ma print(a)	tth.pi) Definition of pi "math.py pi = 3.141592653559783	∦ 36∖python.exe) >
	a = math.cos(ma print(a)	tth.pi) Definition of pi " math.py = < Python 3.60 (E\Anaconda\envs\python p1 = 3.141592653569793	∦ 36∖python.exe) >
	a = math.cos(ma print(a)	tth.pi) Definition of pi " math.py " C Python 3.6.0 (E\Anaconda\envx\python p1 = 3.141592653569793	∦ 36∖python.exe) >
	<pre>a = math.cos(ma print(a)</pre>	Definition of pi "math.py" 1 = 3.141592653509793	,∉ 36\python.exe) >

Vous pouvez obtenir un accès complet au fichier où l'élément est défini. Pour cela, utilisez le raccourcis **CTRL + Clic gauche** sur Windows (ou **CMD + Clic gauche** sur MAC) en cliquant sur l'élément souhaité. Le fichier contenant la définition de l'élément s'ouvrira dans l'éditeur de texte. Cela est très utile si vous voulez retrouver et changer la valeur d'une variable ou la définition d'une fonction.

Vous pouvez faire en sorte de voir la définition d'un élément en passant votre souris sur celuici. Pour cela, allez dans l'onglet **File > Settings**, puis dans l'onglet **Editor > General**. Ici, cochez la case **Show quick documentation on mouse over**.

PC Settings		×
Q. Appearance & Behavior Keymap	Editor > General R Editor > General R Soft Wrans	eset
	Soft Wraps Use soft wraps in editor Use original line's indent for wrapped parts Additional shift: Image: Show soft wrap indicators for current line only Virtual Space Allow placement of caret after end of line Allow placement of caret inside tabs Show writual space at file bottom Other Strip trailing spaces on Save: Ensure line feed at file end on Save Show quick documentation on mouse move Delay (ms): Show quick documentation on mouse move Delay (ms): Show fully platement Different color for lines with whitespace-only modifications Highlight on Caret Movement Highlight on Caret Movement Highlight current scope	

31. **Refactoring** Pycharm permet d'utiliser des refactors, qui permettent de modifier facilement votre code. Pour cela, faites un clic droit sur un élément et affichez les **Refactors**. Selon l'élément sur lequel vous cliquez, différents refractors seront disponibles. Sur une variable, vous pouvez par exemple la renommer : cela la changera dans tous votre script.

Ceci est un programme	de demo!		
A			
import math			
phrase = "Hello world!			
print(phrase)			
for i in range (len (ph)			
print(i*" "+phrase	Copy Reference	Ctrl+Alt+Maj+C	
	Daste	Ctrl+V	
a = math.cos(math.pi)	Paste from History	Ctrl+Maj+V	
print(a)	Paste Simple	Ctrl+Alt+Maj+V	
	Column Selection Mode	Alt+Maj+Insérer	
	Find <u>U</u> sages	Alt+F7	
	<u>R</u> efactor	Þ	Rename Maj+F6
	Folding	•	Change Signature Ctrl+F6
	Go To	•	Move F6
	Generate	Alt+Insérer	С <u>о</u> ру F5
	Run 'Demo'	Ctrl+ Mai+ E10	Safe <u>D</u> elete Alt+Supprimer
	Dahua (Dama)	Cultiviaj+110	Extract
	M Debug Demo		Inline Ctrl+Alt+N
	Kun Demo with Coverage		-
	Profile 'Demo'		Pull Members Up
	Concurrency Diagram for '	Demo'	Push Members Down
	🟓 Save 'Demo'		Convert to Python Package
	Local History		Convert to Python Module