

# 1 Présentation

L'utilisation de `PythonTeX` permet d'insérer du code Python dans le fichier source `LATEX` et de récupérer le résultat dans le texte.

Exemple 1 :

```
Le résultat de 2+2 est \py{2+2}.  
donnera : Le résultat de 2+2 est 4.
```

La possibilité d'utiliser du code python étend les possibilités de codage à toutes les bibliothèques Python.

Exemple 2 :

```
\begin{pycode}  
import numpy as np  
print ("Tirage au sort d'un entier inferieur a dix plus $ \sqrt{2} $ arrondi a 2 decimales  
donne :")  
print (np.round(np.random.randint(10)+np.sqrt(2),2))  
\end{pycode}
```

donnera : *Tirage au sort d'un entier inferieur a dix plus  $\sqrt{2}$  arrondi a 2 decimales donne : 5.41*

Pour plus d'informations, on peut lire le Quickstart ou bien la documentation du package.

## 2 Utilisation

### 2.1 installation

Classique, depuis le CTAN (Il peut être avantageux d'ajouter le répertoire d'installation au `PYTHONPATH` ).

### 2.2 Utilisation *classique*

L'utilisation de `PythonTeX` demande 3 *compilations* :

1. `pdflatex foo.tex`
2. `python pythontex.py foo.tex`
3. `pdflatex foo.tex`

La dernière étape récupère et insère dans le document `LATEX` les résultats de l'exécution du code Python. Si le code Python n'est pas modifié, la première étape suffit.

### 2.3 Utilisation de `PythonTeX` et d'AMC

AMC, depuis la version **r1495** (en développement au 01/12/2013), permet l'utilisation de `PythonTeX` :

1. Il faut ajouter un script nommé **prePythonTex4AMC** dans le répertoire du projet et le rendre exécutable.  
Le script `prePythonTex4AMC` contient les lignes suivantes, en ayant pris soin d'adapter `/chemin/vers/pythontex.py` :

```
#!/bin/sh  
SRC=$1  
echo "***** JOBNAME=$AMC_JOBNAME"  
echo "***** CMD=$AMC_CMD"  
rm -rf ./pythontex-files-*  
rm -f $AMC_JOBNAME.pytxcode  
$AMC_CMD  
echo "***** PYTHON"  
python /chemin/vers/pythontex.py $AMC_JOBNAME.pytxcode  
echo "***** END FILTER"
```

2. Enfin, il faut ajouter au début du fichier `LATEX` les trois lignes suivantes :

```
%%AMC:preprocess_command=prePythonTex4AMC  
%%AMC:jobspecific=1  
%%AMC:latex_engine=pdflatex -shell-escape
```

3. la compilation depuis AMC (par le bouton [Mettre à jour les documents](#)) réalise automatiquement les trois compilations.

### 3 Exemple

```
%%AMC:preprocess_command=prePythonTex4AMC
%%AMC:jobspecific=1
%%AMC:latex_engine=pdflatex --shell-escape
\documentclass{article}
\usepackage[utf8x]{inputenc}
\usepackage{automultiplechoicex}
\usepackage{pythontex}

\begin{document}
\element{Questions}{%
\begin{question}{QA}
La valeur de  $\pi$  est
\begin{reponses}
\bonne {\py{np.pi}}
\mauvaise{\py{100+200}}
\mauvaise{\py{np.random.randint(10)+3.15}}
\end{reponses}
\end{question}

\begin{question}{QB}
\pyc{a=np.random.randint(10)}
\pyc{b=np.random.randint(10)}
\pyc{z=np.complex(a,b)}
Soit le nombre complexe  $\underline{Z}=\py{a}+i.\py{b}$ , quelle est la valeur de  $\underline{Z}^2$ ?
\begin{reponses}
\bonne {\py{z**2}}
\mauvaise{\py{np.conj(z**2)}}
\mauvaise{\py{1/z}}
\end{reponses}
\end{question}
}%

\begin{pycode}
import numpy as np
\end{pycode}

\exemplaire{10} { \restituegroupe{QuestionsIndependantes} }
\end{document}
```