

Notes sur les Docstrings en Python

Les docstrings en Python sont des chaînes de caractères placées immédiatement après la déclaration d'un module, d'une classe, d'une fonction ou d'une méthode. Elles servent à documenter le code et sont accessibles via l'attribut **doc** ou la fonction `help()`.

Exemple :

```
def my_function():  
    '''Demonstrates triple double quotes  
    docstrings and does nothing really.'''  
  
    return None
```

Pour consulter la documentation, on peut utiliser :

```
print(my_function.__doc__)
```

ou :

```
help(my_function)
```

Par convention, les docstrings sont encadrées de triple guillemets et devraient être concises, avec un style one-liner suivi d'une explication plus détaillée.

Exemple PEP257 :

```
def complex(real=0.0, imag=0.0):  
    """Form a complex number.  
  
    Keyword arguments:  
    real -- the real part (default 0.0)  
    imag -- the imaginary part (default 0.0)  
    """
```

La documentation joue un rôle crucial dans la compréhension du code. Par exemple, numpy est largement populaire en partie grâce à sa documentation claire et complète.

Principes à retenir :

- WORO (Write Once, Read Often) : Écrire la documentation une fois pour qu'elle puisse être lue plusieurs fois.
- Les docstrings permettent de générer des simili-tests unitaires avec la librairie doctest.

Exemple d'utilisation de doctest :

```
if __name__ == "__main__":  
    import doctest  
    doctest.testmod()
```

Modèle de docstring :

```
'''  
phrase qui décrit le plus simplement possible  
  
détails.  
exceptions possibles  
exemple (doc test -> simili test unitaire)  
'''
```

L'attribut **doc** est une variable interne d'une fonction, classe ou module contenant la docstring.

Plus d'info, provenant de **ChatGPT** :

Types d'annotations :

Avec l'introduction des annotations de type dans Python 3, il est possible d'enrichir davantage les docstrings en spécifiant les types attendus pour les paramètres et les valeurs de retour des fonctions.

Exemples d'utilisation :

Il est recommandé de garder les docstrings à jour et de les réviser chaque fois que des modifications sont apportées au code correspondant. Cela garantit que la documentation reste précise et utile.

Clarté et Concision :

Les docstrings doivent être claires, concises et faciles à comprendre. Elles doivent fournir suffisamment d'informations pour permettre à l'utilisateur de

comprendre rapidement le but et le fonctionnement de la fonction, de la classe ou du module documenté.

Utilisation de Triple Quotes :

Utilisez des triple quotes (`'''` ou `"""`) pour encadrer les docstrings. Cela permet d'inclure des sauts de ligne et de formater le texte de manière plus lisible.

Description du But :

Décrivez brièvement le but de la fonction, de la classe ou du module documenté dans la première ligne de la docstring. Utilisez un style one-liner pour cette description.