



- Présentation générale
- QUESTIONNAIRE : VOS ATTENTES ET MOTIVATIONS

- Semaine 1. Introduction au MOOC et aux outils Python
- Semaine 2. Notions de base pour écrire son premier programme en Python
- ▼ **Semaine 3. Renforcement des notions de base, références partagées**

1. Les fichiers

Quiz Échéance le janv 25, 2018 at 23:30 UTC

2. Les tuples

Quiz Échéance le janv 25, 2018 at 23:30 UTC

3. Tables de hash

Quiz Échéance le janv 25, 2018 at 23:59 UTC

4. Les dictionnaires

Quiz Échéance le janv 25, 2018 at 23:59 UTC

5. Les ensembles

QUIZ 18 - INTRODUCTIONS AUX CLASSES

(3 points possibles)

Classes et instances

Supposons que nous ayons le code suivant.

```
class Parser:
    def __init__(self, sep):
        self.sep = sep
        self.parsed_line = []

    def parse(self, line):
        self.parsed_line = [i.strip() for i in
                             line.split(self.sep)
                             if i.strip().isdigit()]

    def __str__(self):
        return ' '.join(self.parsed_line)

test = '123 : fj356:34:fj3d: 707'

p = Parser(':')
p.parse(test)
print(p)
```

Quels objets sont `Parser` et `p` ?

☐ `Parser` est une classe et `p` une chaîne de caractères

☐ `Parser` est un module et `p` une classe

☒ `Parser` et `p` sont des classes

Rechercher un cours



☒ `Parser` est une classe et `p` est une instance ✓

EXPLANATION



9. Les exceptions

Quiz Echéance le janv
25, 2018 at 23:59 UTC

7. Les références

partagées

Quiz Echéance le janv
25, 2018 at 23:59 UTC

8. Introduction aux classes

Quiz Echéance le janv
25, 2018 at 23:59 UTC

- ▶ Semaine 4.
Fonctions et
portée des
variables
- ▶ Semaine 5.
Itération,
importation et
espace de
nommage
- ▶ Semaine 6.
Conception
des classes
- ▶ Semaine 7.
L'écosystème
data science
Python
- ▶ Semaine 8.
Programmation
asynchrone -
asyncio

`Parser()` / donc `p` est une instance de la classe `Parser`.

Que représente le paramètre `self` ?

☒ Une référence vers une instance de la classe `Parser` ✓

☐ Une référence vers la classe `Parser`

☒ Une chaîne de caractères ✗

☐ Un mot clef nécessaire pour définir une méthode dans une classe

EXPLANATION

`self` représente bien une référence vers une instance de la classe `Parser`. Cette référence est automatiquement celle de l'instance qui appelle les méthodes de la classe, explicitement comme `parse()` ou implicitement comme `__init__` (lors de la création de l'instance) ou `__str__` (lors d'un `print` de l'instance).

Que va afficher `print(p)` ?

☐ Une exception

☒ `123 34 707` ✓

☒ `None` ✗

☐ Une chaîne de caractères vide

EXPLANATION

On commence par créer un objet, qui contient comme information le séparateur (dans `self.sep`)



noter correspondance des entiers `if i.isdigit():` // on notera que toutes ces opérations peuvent être faites en une seule compréhension de liste.

Enfin avec `print` on invoque en réalité la méthode `__str__`, qui tire profit du calcul de `parse` (rangé dans `self.parsed_line`) pour formater, cette fois avec un espace comme séparateur.

Vous avez utilisé 3 essais sur 3

[A propos](#)

[Aide](#)

[Contact](#)

[Conditions générales d'utilisation](#)

[Charte utilisateurs](#)

[Politique de confidentialité](#)

[Mentions légales](#)



POWERED BY
OPENedX