

1. Qu'est-ce que NumPy ?

Réponse :

NumPy (Numerical Python) est une bibliothèque Python open source utilisée pour effectuer des calculs numériques efficaces. Elle permet de manipuler des tableaux multidimensionnels (arrays) et fournit de nombreuses fonctions mathématiques optimisées.

2. Pourquoi utiliser NumPy au lieu des listes Python ?

Réponse :

NumPy est plus rapide et plus efficace en mémoire que les listes Python classiques. Il permet d'effectuer des opérations vectorisées (sans boucle explicite), ce qui rend les calculs plus rapides et le code plus concis.

3. Qu'est-ce qu'un array NumPy ?

Réponse :

Un array NumPy est une structure de données similaire à une liste, mais optimisée pour les calculs numériques. Il peut être unidimensionnel, bidimensionnel (comme une matrice), ou multidimensionnel.

4. Comment créer un tableau NumPy ?

Réponse :

On peut créer un tableau avec la fonction `np.array()` :

```
import numpy as np  
a = np.array([1, 2, 3])
```

Il existe aussi d'autres fonctions comme `np.zeros()`, `np.ones()`, `np.arange()`, ou `np.linspace()`.

5. Quelle est la différence entre une liste Python et un tableau NumPy ?

Réponse :

- Les listes peuvent contenir des types différents ; les arrays NumPy sont typés.
- Les arrays sont plus rapides et plus légers en mémoire.
- Les opérations mathématiques s'appliquent directement sur les arrays, pas sur les listes.

6. Quelles sont les fonctions mathématiques disponibles dans NumPy ?

Réponse :

NumPy fournit des fonctions comme :

- `np.mean()` (moyenne),
- `np.std()` (écart type),
- `np.sum()`, `np.min()`, `np.max()`,
- `np.dot()` (produit scalaire),
- `np.linalg.inv()` (matrice inverse), etc.

7. Qu'est-ce que le broadcasting dans NumPy ?

Réponse :

Le broadcasting est un mécanisme qui permet d'effectuer des opérations entre tableaux de formes différentes, en étendant automatiquement les dimensions si nécessaire.

8. Quelle est la différence entre `np.array` et `np.matrix` ?

Réponse :

`np.array` est plus général et recommandé. `np.matrix` est une classe plus ancienne, limitée à deux dimensions, et dont l'usage est désormais déconseillé.

9. Comment accéder à un élément d'un tableau NumPy ?**Réponse :**

Comme avec les listes :

```
a = np.array([10, 20, 30])
print(a[1])  # Affiche 20
```

Pour les matrices :

```
b = np.array([[1, 2], [3, 4]])
print(b[1, 0])  # Affiche 3
```

10. Comment installer NumPy ?**Réponse :**

Avec pip :

```
pip install numpy
```