

Erreurs de type python jour-15

Types d'erreur Python

Lorsque nous écrivons du code, il est courant que nous créons une faute de frappe ou une autre erreur courante. Si notre code ne fonctionne pas, l'interpréteur Python affichera un message, contenant des commentaires avec des informations sur l'endroit où le problème se produit et le type d'erreur. Il nous donne également parfois des suggestions sur une solution possible. Comprendre différents types d'erreurs dans les langages de programmation nous aidera à déboguer notre code rapidement et cela nous rend mieux dans ce que nous faisons.

Voyons les types d'erreur les plus courants un par un. Ouvrons d'abord notre coque interactive Python. Accédez à votre terminal informatique et écrivez «Python». La coque interactive Python sera ouverte.

Syntaxerror

Exemple 1: syntaxerror

```
asabeneh @ asabeneh: ~ $ python python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Help", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> Print Fichier 'Hello World' "< stdin >", ligne 1 imprimé 'Hello World'
```

```
^
```

```
SyntaxError: parenthèses manquantes en appel à «imprimer». Veuillez-vous dire imprimer ('Hello World')?
```

```
>>>
```

Comme vous pouvez le voir, nous avons fait une erreur de syntaxe parce que nous avons oublié d'enfermer la chaîne avec parenthèse et Python suggère déjà la solution. Laissez-nous le réparer.

```
asabeneh @ asabeneh: ~ $ python python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Help", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> Print Fichier 'Hello World' "< stdin >", ligne 1 imprimé 'Hello World' ^
```

```
SyntaxError: parenthèses manquantes en appel à «imprimer». Veuillez-vous dire imprimer ('Hello World')? >>> print ('Hello World')
```

```
Bonjour Monde
>>>
```

L'erreur était un **SyntaxError**. Après le correctif, notre code a été exécuté sans accroc. Voir plus de types d'erreur.

Noméror

Exemple 1: NameError

```
asabeneh @ asabeneh: ~ $ python python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Aide", "Copyright", "Crédits" ou "Licence" pour plus d'informations. >>> print (age)
Traceback (dernier appel dernier):
Fichier "< stdin >", ligne 1, dans le module < > nameError: name 'Age' n'est pas défini
>>>
```

Comme vous pouvez le voir sur le message ci-dessus, l'âge du nom n'est pas défini. Oui, il est vrai que nous n'avons pas défini de variable d'âge, mais nous essayons de l'imprimer comme si nous l'avions déclaré. Maintenant, permet de résoudre ce problème en le déclarant et en affectant avec une valeur.

```
Asabeneh @ Asabeneh: ~ $ Python Python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] Sur Darwin Type "Aide", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> print (age)
Traceback (Call le plus récent Last):
Fichier "< stdin >", ligne 1, dans < module > nameError: name 'Age' n'est pas défini
>>> Age = 25 >>> print (Age) 25 >>>
```

Le type d'erreur était un **NameError**. Nous avons débogué l'erreur en définissant le nom de la variable.

Indexerror

Exemple 1: IndexError

```
asabeneh @ asabeneh: ~ $ python python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Help", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> Nombres = [1, 2, 3, 4, 5] >>> Nombres[5] Traceback (Call le plus récent Last): Fichier "< STDIN >", ligne 1, dans < module > IndexError: Liste d'index de la plage >>>
```

Dans l'exemple ci-dessus, Python a soulevé un ***IndexError***, car la liste n'a que des index de 0 à 4, il était donc hors de portée.

ModuleNotFoundError

Exemple 1: moduleNotFoundError

```
Asabeneh @ Asabeneh: ~ $ Python Python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Help", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> Importer Maths Traceback (Call le plus récent Last): Fichier "< stdin >", ligne 1, dans < module > moduleNotFoundError: pas de module nommé 'maths' >>>
```

Dans l'exemple ci-dessus, j'ai ajouté un supplément en mathématiques délibérément et ***ModuleNotFoundError*** a été soulevé. Permet de le réparer en supprimant les S supplémentaires des mathématiques.

```
asabeneh @ asabeneh: ~ $ python python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Help", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> Importer Maths Traceback (Call le plus récent Last): Fichier "< stdin >", ligne 1, dans < module > moduleNotFoundError: pas de module nommé 'maths' >>> Import Math Math Math Math Math Math Math:
```

```
>>>
```

Nous l'avons corrigé, alors utilisons certaines des fonctions du module mathématique.

AttributError

Exemple 1: attributError

```
asabeneh @ asabeneh: ~ $ python python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Help", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> import maths Traceback (most recent call last):
File "<stdin>", line 1, in <module> ModuleNotFoundError: No module named 'maths'
>>> import math >>> math.PI Traceback (most recent call last): File "<stdin>", Ligne 1, dans le module < > AttributeError: Module 'Math' n'a pas d'attribut 'pi' >>>
```

Comme vous pouvez le voir, j'ai fait une erreur à nouveau! Au lieu de PI, j'ai essayé d'appeler une fonction PI à partir du module mathématique. Il a soulevé une erreur d'attribut, cela signifie que la fonction n'existe pas dans le module. Permet de le réparer en passant de Pi à Pi.

```
Asabeneh @ Asabeneh: ~ $ Python Python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Help", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> Importer maths traceback (dernier appel dernier): fichier "< stdin >", ligne 1, dans < module > modulenotFounDerror: pas de module nommé "maths" >>> importath >>> math.pi traceback (le plus récent appel): Fichier " < stdin >", ligne 1, dans < module > attributError: module 'math' n'a pas d'attribut 'pi'
>>> math.pi 3.141592653589793 >>>
```

Maintenant, lorsque nous appelons Pi du module mathématique, nous avons obtenu le résultat.

Merle

Exemple 1: Keyerror

```
asabeneh @ asabeneh: ~ $ python python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Help", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> utilisateurs = {'name': 'asab', 'age': 250, 'country': 'Finland'} >>> utilisateurs['name'] 'asab' >>> utilisateurs['comté'] trace (le plus récent appel dernier): Fichier "< stdin >", ligne 1, dans < stdin >", ligne 1, dans < module {V9} KeyError: 'comté' >>>
```

Comme vous pouvez le voir, il y avait une faute de frappe dans la clé utilisée pour obtenir la valeur du dictionnaire. Il s'agit donc d'une erreur clé et le correctif est assez simple. Faisons ça!

```
Asabeneh @ Asabeneh: ~ $ Python Python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Help", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> user = {'name': 'Asab', 'age': 250, 'country': 'Finland'} >>> user['name'] 'Asab' >>> user['county'] Traceback (most recent call last): File "< stdin >", line 1, in < Module > KeyError: 'comté' >>> utilisateur['country'] 'Finlande' >>>
```

Nous avons débogué l'erreur, notre code s'est déroulé et nous avons obtenu la valeur.

Mât de type

Exemple 1: TypeError

```
asabeneh @ asabeneh: ~ $ python python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Help", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> 4 + '3' Traceback (Call le dernier dernier): fichier "< stdin >", ligne 1, dans < module > TypeError: UNSUPPORTED Operand Type (S) pour +: 'int' et 'str' >>>
```

Dans l'exemple ci-dessus, un type EERROR est soulevé car nous ne pouvons pas ajouter un nombre à une chaîne. La première solution serait de convertir la chaîne en int ou float. Une autre solution

convertirait le nombre en une chaîne (le résultat serait alors «43»). Suivons le premier correctif.

```
asabeneh @ asabeneh: ~ $ python python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Help", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> 4 + '3'
Traceback (dernier appel dernier): fichier "<stdin>", ligne 1, dans <module> TypeError: non supporté d'opérands (s) pour +: 'int' et 'str'
>>> 4 + 4 + float('3')
7.0 >>>
```

Erreur supprimée et nous avons obtenu le résultat que nous attendions.

Importateur

Exemple 1: TypeError

```
Asabeneh @ Asabeneh: ~ $ Python Python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Help", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> de Math Import Power
Traceback (Call le plus récent Last): Fichier "<stdin>", ligne 1, dans <module> Importror: ne peut pas importer le nom 'Power' de 'Math'
>>>
```

Il n'y a aucune fonction appelée puissance dans le module mathématique, il va avec un nom différent: **pow**. Corrigons-le:

```
asabeneh @ asabeneh: ~ $ python python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Help", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> à partir de Math Import Power
Traceback (Call le plus récent Last): Fichier "<stdin>", ligne 1, dans <module> Importror: Importation de Name 'Power' de 'Math'
>>> à partir de Math Import Pow >>> Pow (2,3)
8.0 >>>
```

Value Error

```
asabeneh @ asabeneh: ~ $ python python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Help", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> int('12a')
Traceback (dernier appel dernier): fichier "<stdin>", ligne 1, dans <module> ValueError: invalid littéral pour int () avec la base 10: '12a' >>>
```

Dans ce cas, nous ne pouvons pas changer la chaîne donnée en un nombre, en raison de la lettre «A».

Zérodivisionerror

```
Asabeneh @ Asabeneh: ~ $ Python Python 3.9.6 (par défaut, 28 juin 2021, 15:26:21) [Clang 11.0.0 (Clang-1100.0.33.8)] sur Darwin Type "Aide", "Copyright", "crédits" ou "Licence" pour plus d'informations. >>> 1/0
Traceback (Call le plus récent Last): Fichier "<stdin>", ligne 1, dans <module> ZeroDivisionError: Division par zéro >>>
```

Nous ne pouvons pas diviser un nombre par zéro.

Nous avons couvert certains des types d'erreur Python, si vous souhaitez en vérifier plus, vérifiez la documentation Python sur les types d'erreur Python. Si vous êtes doué pour lire les types d'erreur, vous pourrez réparer vos bogues rapidement et vous deviendrez également un meilleur programmeur.

Vous excellez. Vous vous êtes rendu à mi-chemin jusqu'à votre chemin vers la grande ur. Maintenant, faites des exercices pour votre cerveau et pour vos muscles.

Exercices: Jour 15

1. Ouvrez votre shell interactif Python et essayez tous les exemples couverts dans cette section.

🎉 CONGRATULATIONS ! 🎉