



Emmad Karim

19 août 2021, 00:22

Actuellement, les paramètres que j'utilise avec NotePad++ donnent cette vue

```

50 IDENTIFICATION DIVISION.
51 PROGRAM-ID. HELLO-WORLD.
52 DATA DIVISION.
53 WORKING-STORAGE SECTION.
54 01 WS-MSG PIC X(100).
55 01 P1 PIC S9(4) COMP.
56 01 P2 PIC S9(4) COMP.
57 01 ERR-REC.
58     02 ERR-L          PIC X(10).
59     02 F              PIC X VALUE ":".
60     02 ERR-TXT1       PIC X(50).
61     02 SET1.
62         03 ERR-VAL1-N PIC +Z(22)VZ(7).
63         03 ERR-TXT2   REDEFINES ERR-VAL1-N PIC X(30).
64     02 SET2.
65         03 ERR-VAL2-N PIC +Z(22).Z(7).

```

Mais j'aimerais avoir cette vue

```

32 PROCEDURE DIVISION.
33 DISPLAY 'Hello, world' ERR-REC.
34
35 MOVE "AAA ?1 BBB ?2 CCC ?3" TO WS-MSG
36 MOVE 10 TO ENX9 (1)
37 MOVE "[TEN]" TO ENXT(2)
38 MOVE -15.33 TO ENX9(3)
39
40 MOVE ZERO TO KK.
41 UNSTRING WS-MSG INTO SUB-MSG(1)
42                      SUB-MSG(2)
43                      SUB-MSG(3)
44                      SUB-MSG(4)
45                      SUB-MSG(5)
46                      TALLYING KK
47                      DELIMITED "?"
48
49 UNSTRING LS-IN-TXT DELIMITED BY LS-IN-DELIMITER
50                      INTO LS-FIELD-NAME  COUNT A-CNT,
51                      LS-FIELD-VALUE  COUNT B-CNT,
52                      C  COUNT C-CNT
53                      TALLYING K
54

```



18 sur 18



Je dois également ajouter des tabulations aux colonnes 7, 8, 12 et 72 (en espérant que la tabulation s'affiche comme une colonne verticale).

Je ne sais rien de la personnalisation, des polices, etc. Quelqu'un pourrait-il me donner un conseil ?

J'aime particulièrement la police ci-dessus. Par où commencer ? Que faire ?

 2 réponses Dernière réponse 19 août 2021, 21:00 ▼

^ 0 v ⋮



**Écopalypse**

19 août 2021, 10:52

Npp fournit ce que l'on appelle des thèmes.

Voir ici [pour plus d'informations](#) à ce sujet. Si quelque chose n'est pas clair, faites-le nous savoir.

^ 0 v ⋮



**Emmad Karim**

19 août 2021, 11:29

Connaissez-vous un thème pour le langage COBOL ? Merci.

 2 réponses Dernière réponse 19 août 2021, 11:31 ▼

^ 0 v ⋮



**Écopalypse**  @Emmad Kareem

19 août 2021, 11:31

@Emmad-Kareem

Il y a un malentendu, un thème n'est pas pour une langue mais pour toutes. Npp en propose tout un tas, lesquels avez-vous essayés ?

^ 0 v ⋮



**Emmad Karim**  @Ekopalypse

19 août 2021, 13:50

@Ekopalypse Merci.

^ 0 v ⋮



**Michel Vincent**  @Emmad Kareem

19 août 2021, 15:13

@Emmad-Kareem a dit dans Personnalisation pour COBOL :

*Connaissez-vous un thème pour le langage COBOL ? Merci.*



18 sur 18



Comme le dit @Ekopalypse , il existe de nombreux thèmes pour « toutes » les langues, et non un thème spécifique pour une langue spécifique. Cela dit, de nombreux thèmes du répertoire Notepad++/themes n'ont pas été mis à jour depuis un certain temps et ne reflètent donc pas les derniers changements/ajouts de langues dans Notepad++. Par exemple, j'utilise un thème personnalisé basé sur le thème Oblivion et en vérifiant cette question, j'ai réalisé que mon thème n'avait pas le langage Cobol - donc pas de style.

Choisissez le thème que vous souhaitez. Si les fichiers Cobol n'ont pas de style, le langage n'est probablement pas inclus dans votre thème.

1. Fermer Notepad++
2. Ouvrez le fichier de thème Notepad++/themes/YOUR\_THEME\_FILE.xml avec le Bloc-notes Windows et ouvrez le fichier Notepad++/stylers.xml.
3. Copiez la section concernée Notepad++/stylers.xml dans Notepad++/themes/YOUR\_THEME\_FILE.xml et enregistrez.
4. Rouvrez Notepad++ et ouvrez votre fichier Cobol. Il devrait être stylisé, probablement pas à votre goût.
5. Utilisez les Paramètres => Configurateur de style pour ajuster les couleurs de votre langue (Cobol).

Pour commencer, voici la section Cobol pertinente de mon thème :

```
<LexerType name="cobol" desc="COBOL" ext="">
  <WordsStyle name="PREPROCESSOR" styleID="9" fgColor="A082BD" bgColor="
  <WordsStyle name="DEFAULT" styleID="11" fgColor="E0E2E4" bgColor="2931
  <WordsStyle name="DECLARATION" styleID="5" fgColor="A082BD" bgColor="2
  <WordsStyle name="INSTRUCTION WORD" styleID="16" fgColor="93C763" bgCo
  <WordsStyle name="KEYWORD" styleID="8" fgColor="93C763" bgColor="29313
  <WordsStyle name="NUMBER" styleID="4" fgColor="FFCD22" bgColor="293134
  <WordsStyle name="STRING" styleID="6" fgColor="EC7600" bgColor="293134
  <WordsStyle name="CHARACTER" styleID="7" fgColor="FF8409" bgColor="293
  <WordsStyle name="OPERATOR" styleID="10" fgColor="E8E2B7" bgColor="293
  <WordsStyle name="COMMENT" styleID="1" fgColor="66747B" bgColor="29313
  <WordsStyle name="COMMENT LINE" styleID="2" fgColor="66747B" bgColor="
  <WordsStyle name="COMMENT DOC" styleID="3" fgColor="66747B" bgColor="2
  <WordsStyle name="COMMENT LINE DOC" styleID="15" fgColor="66747B" bgCo
  <WordsStyle name="COMMENT DOC KEYWORD" styleID="17" fgColor="66747B" b
  <WordsStyle name="COMMENT DOC KEYWORD ERROR" styleID="18" fgColor="FF4
</LexerType>
```

Acclamations.

1 réponse Dernière réponse 21 sept. 2022, 00:09

3



Alan Kilborn

19 août 2021, 15:42

@Emmad-Kareem a dit dans Personnalisation pour COBOL :

Je dois également ajouter des tabulations aux colonnes 7, 8, 12 et 72



18 sur 18



Jusqu'à présent, cette partie de la question de l'OP n'a pas été prise en compte par les répondants. Je ne me souviens pas si cette question a déjà été soulevée, mais il semble bien que ce soit le cas.

Notepad++/Scintilla peut-il gérer les positions de tabulation à largeur variable ?

 2 réponses Dernière réponse 19 août 2021, 16:33 ▼

^ 1 v ⋮



**PeterJones** @Alan Kilborn

 19 août 2021, 16:33

@Alan-Kilborn a dit dans Personnalisation pour COBOL :

*Notepad++/Scintilla peut-il gérer les positions de tabulation à largeur variable ?*

voir la discussion <https://community.notepad-plus-plus.org/topic/18429/need-to-set-tab-stops-not-spacing> d'il y a quelques années. @Ekopalypse a pu obtenir le

SCI\_CLEARABSTOPS/SCI\_ADDTABSTOP  messages via l'interface PythonScript

`editor.clearTabStops()/ editor.addTabStop()` pour configurer un ensemble personnalisé d'emplacements d'onglets. Malheureusement, cela se fait sur une base par ligne et doit être réappliqué fréquemment (à chaque fois que le tampon est mis à jour). De plus, cela le définit en pixels, pas en caractères, vous devez donc effectuer une sorte de facteur de conversion pour convertir l'emplacement des caractères en pixels.

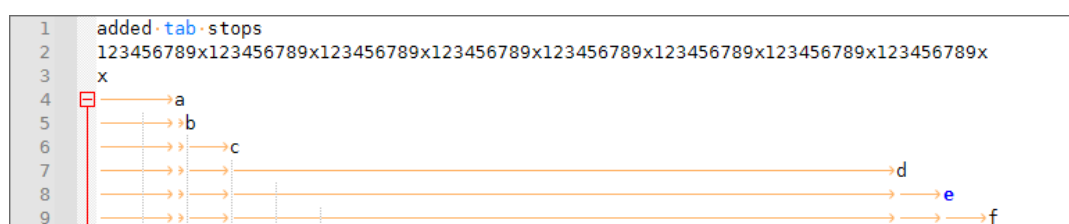
Je suppose que cela devrait ressembler à ce qui suit. Veuillez noter qu'il s'agit simplement d'une preuve de concept et qu'elle doit être traitée comme du code qui nécessiterait un débogage et un travail supplémentaire pour le rendre à 100 %.

```
# encoding=utf-8
"""in response to https://community.notepad-plus-plus.org/topic/21689/"""
from Npp import *

cols = (7,8,12, 72)
pixperchar = 8

def set_tab_via_callback():
    for linenum in range(editor.getLineCount()):
        editor.clearTabStops(linenum)
        for tabcol in cols:
            editor.addTabStop(linenum, tabcol*pixperchar)

set_tab_via_callback()
```



18 sur 18

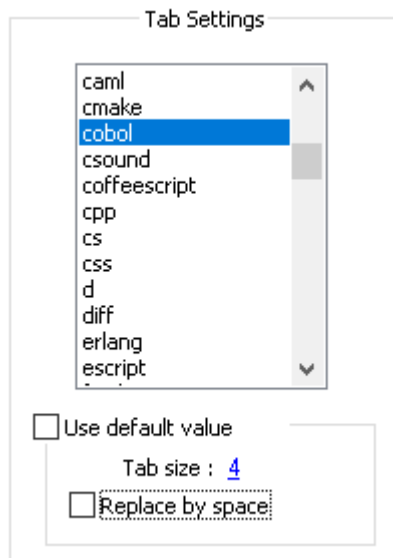


Au lieu d'exécuter manuellement la fonction `set_tab_via_callback()`, vous devrez ajouter des rappels pour cette fonction ; le message d'Eko lié précédemment indique quels rappels sont nécessaires ; vous devrez également calculer ou coder en dur le droit `pixperchar` de convertir correctement les caractères en pixels pour votre police ; vous voudrez probablement aussi vérifier le rappel pour vous assurer qu'il n'exécute le changeur d'onglets que si le fichier actif est COBOL.

Peut-être qu'Eko a toujours l'exemple de code avec lequel il a joué (mais qui n'a pas été publié) dans ce sujet précédent.

----

**PS (modifier)** : j'ai oublié de mentionner que je devais aller dans Paramètres > Préférences > Langue > Paramètres des onglets, et configurer `cobol` pour *ne pas* remplacer par un espace :



^ 4 v ⋮



Écopalypse

19 août 2021, 17:49

Ici [c'est ce que j'ai posté sur github](#) concernant la largeur de l'onglet spécifique à la langue.

^ 2 v ⋮



PeterJones ↩@Ekopalypse

19 août 2021, 18:03

@Ekopalypse a dit dans Personnalisation pour COBOL :

Ici [c'est ce que j'ai posté sur github](#) concernant la largeur de l'onglet spécifique à la langue.

Cela ressemble plus à du multiEdge qu'à des tabulations, à moins que je n'aie mal compris. Mais le même concept de base devrait fonctionner (en remplaçant l'effacement des bords verticaux par une boucle par ligne qui efface les tabulations, et en remplaçant l'ajout de multiEdge par des boucles comme ajouter les tabulations dans une boucle par ligne).



18 sur 18



 2 réponses Dernière réponse 19 août 2021, 18:07

^ 3 v ⋮

**Écopalypse** ↩@PeterJones19 août 2021, 18:07

@PeterJones

tu as raison, c'est ma faute.

^ 0 v ⋮

**PeterJones** ↩@PeterJones19 août 2021, 18:38

Cette fusion du script d'Eko et du mien s'en rapproche... J'ai fusionné ma preuve de concept avec l'enregistrement de wrapper sophistiqué d'Eko pour fournir différentes mises à jour en fonction de chaque langue, puis j'ai vérifié par rapport au [flux suggéré](#) il y a 2 ans

Mais je n'ai pas pu trouver quel(s) message(s) étaient signifiés par le message [suggéré on\\_update\\_ui](#). Je pense qu'il faudrait peut-être une ou plusieurs notifications Scintilla pour cela, donc il faudrait utiliser un message distinct `editor.addCallback()` en plus du message `notepad.addCallback()`

Ainsi, avec cette version, les onglets seront mis à jour à chaque fois que vous passerez à une fenêtre COBOL ou que vous changerez la langue du fichier actif de COBOL à partir d'autre chose.

```
# encoding=utf-8
"""in response to https://community.notepad-plus-plus.org/topic/21689/

Merge my suggestion with Eko's pointer to https://github.com/notepad-plus-plus/not
"""
from Npp import editor, notepad, NOTIFICATION, EDGEVISUALSTYLE

TAB_CONFIG = {}

pixperchar = 8

def register_lang(func):
    def wrapped():
        for linenum in range(editor.getLineCount()):
            editor.clearTabStops(linenum)
            func(linenum)
    TAB_CONFIG[func.__name__.replace('on_lang_', '')] = wrapped
```

^ 1 v ⋮

**PeterJones** ↩@PeterJones 19 août 2021, 19:40

mis à jour avec une notification MODIFIÉE scintilla, avec l'avertissement qu'elle ne se déclenchera que pour `linesAdded` une valeur non nulle - c'est-à-dire non nulle si une ligne a été supprimée.



18 sur 18



c'est un peu lent, mais lorsque vous accédez au fichier ou ajoutez ou supprimez des lignes, tous les onglets seront mis à jour.

```
# encoding=utf-8
"""in response to https://community.notepad-plus-plus.org/topic/21689/

Merge my suggestion with Eko's pointer to https://github.com/notepad-plus-plus/notepad-plus-plus/pull/11111
"""
from Npp import editor, notepad, NOTIFICATION, SCINTILLANOTIFICATION, EDGEVISUALSTOPS

TAB_CONFIG = {}

pixperchar = 8

def register_lang(func):
    def wrapped():
        for linenum in range(editor.getLineCount()):
            editor.clearTabStops(linenum)
            func(linenum)
    TAB_CONFIG[func.__name__.replace('on_lang_', '')] = wrapped
```

^ 1 v ⋮



**Michel Vincent** @Alan Kilborn

19 août 2021, 20:27

@Alan-Kilborn a dit dans Personnalisation pour COBOL :

*Je dois également ajouter des tabulations aux colonnes 7, 8, 12 et 72*

*Jusqu'à présent, cette partie de la question de l'OP n'a pas été prise en compte par les répondants. Je ne me souviens pas si cette question a déjà été soulevée, mais il semble bien que ce soit le cas.*

*Notepad++/Scintilla peut-il gérer les positions de taquets de tabulation à largeur variable ?*

Est-ce que les **TabStops élastiques** de @dail [Le plugin](#) résout-il ce problème ? Honnêtement, je ne sais pas et je ne l'ai pas essayé (je n'en ai jamais eu besoin), mais il semble que cela puisse être le cas ?

Acclamations.

^ 0 v ⋮



**Peter Jones** @Michael Vincent

19 août 2021, 20:52

@Michael-Vincent ,

Je ne l'utilise pas non plus, mais mon impression était que non. Mais je l'ai pris pour le tester brièvement.



18 sur 18



```

7      → indent . and
8  [ tabstops → align . code
9
10     elastic → tabstops
11     longer . word → after
12  [ hello → there
13
14     what . about . a . → new . alignment
15     → here
16  [ there
17
18     what . about → in . → between → asdfas
19     x → hmp → asdfsad
20     whatever → there →

```

les tabulations dans Elastic Tabstops s'étireront pour s'adapter au texte le plus long : par exemple, avec `what about`, `x` et `whatever`, la première tabulation sera suffisamment large pour le `what about`. Et cela permet des arrêts différents lorsque vous donnez une ligne vide entre les paragraphes/sections... mais je ne vois pas comment prédéfinir des largeurs spécifiques.

^ 1 v ⋮



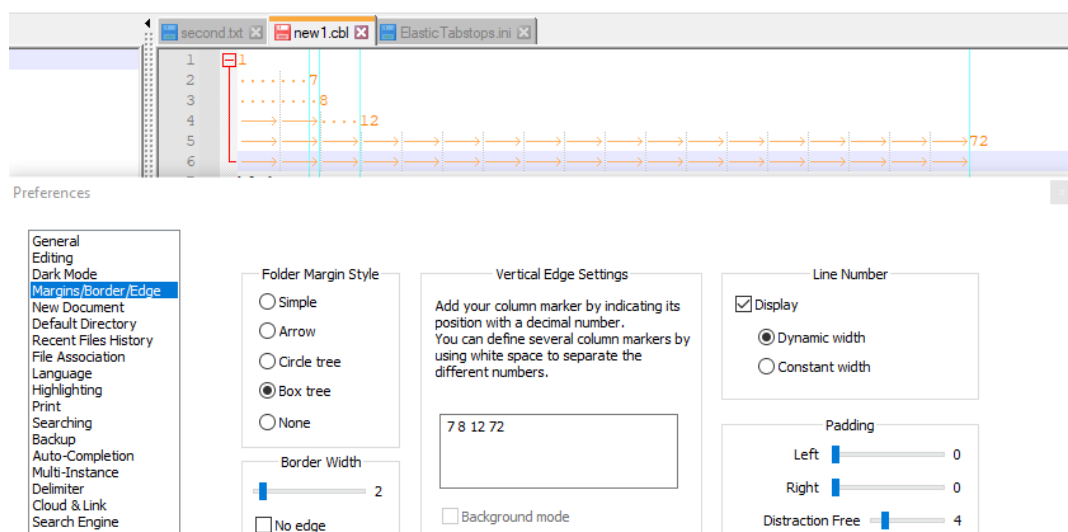
PeterJones @Emmad Kareem

19 août 2021, 21:00

@Emmad-Kareem a dit dans Personnalisation pour COBOL :

*Je dois également ajouter des tabulations aux colonnes 7, 8, 12 et 72 (en espérant que la tabulation s'affichera comme une colonne verticale).*

Je ne pense pas que quelqu'un ait directement abordé ce sujet non plus. Avec une version récente de Notepad++, vous pouvez avoir plusieurs bords verticaux dessinés : allez dans **Paramètres > Préférences > Marges / Bordure / Bordure** et remplissez les **Paramètres de bord vertical** :



Cet exemple n'a pas les rappels Python actifs, donc la taille de l'onglet est une largeur fixe. Le voici avec Elastic Tabstops également actif, avec une ligne d'en-tête factice pour essayer de forcer les

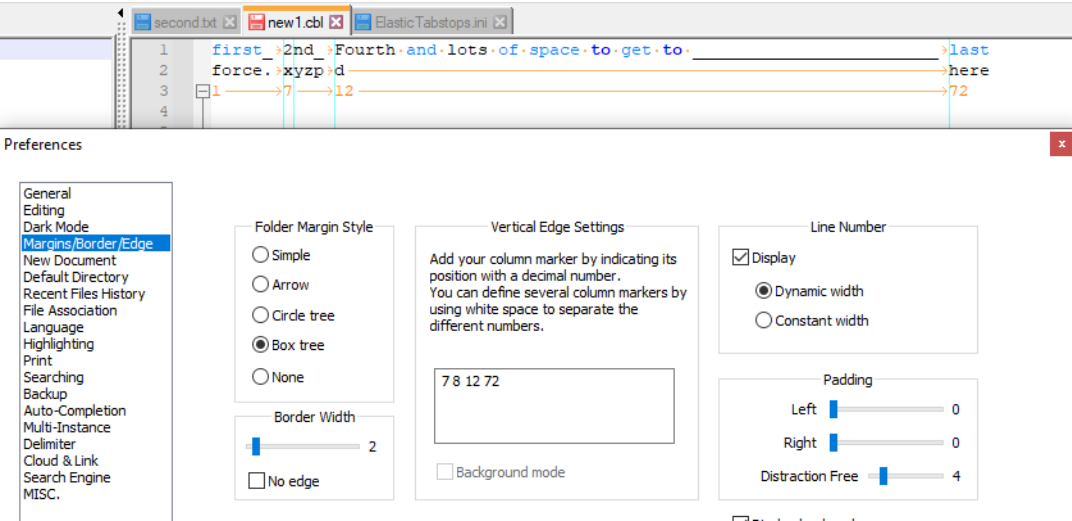


18 sur 18





taquets de tabulation connus :



L'inconvénient des bords verticaux multiples est qu'ils sont identiques pour toutes les langues. Cependant, l'exemple multiEdge auquel @Ekopalypse a fait référence résoudra réellement ce problème pour vous.

En parlant de la taille/emplacement des tabulations, une chose que vous ne mentionnez pas : COBOL accepte-t-il l'utilisation du caractère TAB entre les champs ? Ou avez-vous besoin/souhaitez-vous une conversion de tabulation en espaces (de sorte que lorsque vous appuyez sur TAB, il utilisera réellement suffisamment de caractères d'espace pour combler l'espace) ? Si vous acceptez de conserver TAB comme caractère TAB, alors le PythonScript ou un ElasticTabstops-with-header peut fonctionner pour vous ; si vous avez besoin que cet espacement soit codé sous forme de caractères d'espace réels (ASCII 32 = 0x20) dans le fichier, alors aucun des deux ne fournira ce dont vous avez besoin.

^ 1 v ⋮

environ 8 mois après



Eduardo Martines Teixeira @Emmad Kareem

27 avr. 2022, 23:44



Communauté

# Personnalisation pour COBOL

? Aide recherchée .....

18 messages

6 publieurs

4,4k vues

Se connecter pour répondre

```
<WordsStyle name="DEFAULT" styleID="11" fgColor="DCDCCC" bgColor="3F3F3F"
<WordsStyle name="DECLARATION" styleID="5" fgColor="FFEBDD" bgColor="3F
<WordsStyle name="INSTRUCTION WORD" styleID="16" fgColor="DFC47D" bgCol
<WordsStyle name="KEYWORD" styleID="8" fgColor="FFCF4F" bgColor="3F3F3F
<WordsStyle name="NUMBER" styleID="4" fgColor="8CD0D3" bgColor="3F3F3F"
<WordsStyle name="STRING" styleID="6" fgColor="E3CEAB" bgColor="3F3F3F"
<WordsStyle name="CHARACTER" styleID="7" fgColor="DCA3A3" bgColor="3F3F
<WordsStyle name="OPERATOR" styleID="10" fgColor="
```



18 sur 18



```
<WordsStyle name="COMMENT LINE" styleID="2" fgColor="7F9F7F" bgColor="3F9F7F">
<WordsStyle name="COMMENT DOC" styleID="3" fgColor="7F9F7F" bgColor="3F9F7F">
<WordsStyle name="COMMENT LINE DOC" styleID="15" fgColor="7F9F7F" bgColor="3F9F7F">
<WordsStyle name="COMMENT DOC KEYWORD" styleID="17" fgColor="7F9F7F" bgColor="3F9F7F">
<WordsStyle name="COMMENT DOC KEYWORD ERROR" styleID="18" fgColor="7F9F7F" bgColor="3F9F7F">
</LexerType>
```

code\_text

^ 0 v ⋮

environ 5 mois après



**Emmad Karim** ↩@Michael Vincent

21 sept. 2022, 00:09

@Michael-Vincent , merci beaucoup pour votre aide.

^ 0 v ⋮

La communauté des utilisateurs de l' éditeur de texte Notepad++ .

Propulsé par [NodeBB](#) | [Contributeurs](#)