

# Le blog de codage de Kevin

J'améliore mes compétences en programmation étape par étape. Vous êtes invités !

Menu

## MOM Pt. 003 : Découvrir COBOL – Compilation

👤 kévindurant

📁 Mainframe , mon propre mainframe

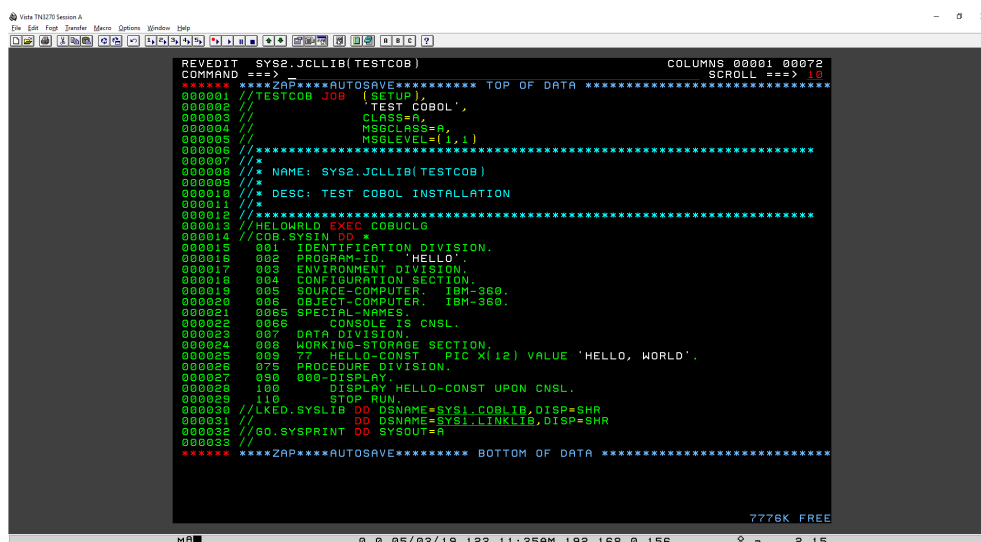
🕒 3 mai 2019

⌵ 2 minutes

Aujourd'hui, je vais découvrir COBOL. J'ai déjà utilisé COBOL dans le cadre du concours Master the Mainframe 2018 d'IBM, mais cette fois, je travaille en MVS 3.8 !

Qu'est-ce que je prévois de faire ici ? Eh bien, avant de commencer à apprendre COBOL, je veux savoir comment je peux l'éditer, le compiler et l'exécuter sur mon propre mainframe. MVS3.8 fournit un petit morceau de code de test dans **SYS2.JCLLIB** appelé **TESTCOB**.

Je vais donc dans **RFE**. Ensuite, je dois naviguer dans l'utilitaire **DSLIST**, ce qui signifie que je saisis **3.4** pour y accéder rapidement. Le préfixe du nom de l'ensemble de données est **SYS2.JCLLIB**. Lors de l'édition, je peux faire défiler vers le bas et éditer **TESTCOB**.



```
REEDIT COMMAND ==> SYS2.JCLLIB( TESTCOB) COLUMNS 00001 00072
*****ZAP*****AUTOSAVE***** TOP OF DATA *****
000001  (SETUP)
000002  TESTCOB JOB  'TEST COBOL',
000003  CLASS=A,
000004  MSGCLASS=A,
000005  MSGLEVEL=(1,1)
000006  *****
000007  * NAME: SYS2.JCLLIB(TESTCOB)
000008  *
000009  * DESC: TEST COBOL INSTALLATION
000010  *
000011  *****
000012  //*****
000013  //HELLOWLD EXEC COBUCLG
000014  //COB.SYSIN DD *
000015  001  IDENTIFICATION DIVISION.
000016  002  PROGRAM-ID. 'HELLO'.
000017  003  ENVIRONMENT DIVISION.
000018  004  CONFIGURATION SECTION.
000019  005  SOURCE-COMPUTER. IBM-360.
000020  006  OBJECT-COMPUTER. IBM-360.
000021  0065 SPECIAL-NAMES.
000022  0066  CONSOLE IS CNL.
000023  007  DATA DIVISION.
000024  008  WORKING-STORAGE SECTION.
000025  009  77 HELLO-CONST PIC X(12) VALUE 'HELLO, WORLD'.
000026  005  PROCEDURE DIVISION.
000027  000 000-DISPLAY.
000028  100  DISPLAY HELLO-CONST UPON CNL.
000029  110  STOP RUN.
000030  //LKED.SYSLIB DD DSN=SYS1.COBLIB,DISP=SHR
000031  //DD DSN=SYS1.LINKLIB,DISP=SHR
000032  //GO.SYSPRINT DD SYSOUT=A
000033  *****ZAP*****AUTOSAVE***** BOTTOM OF DATA *****
7776K FREE
MA 0.0 05/03/19.123 11:35AM 192.168.0.156 2.15
```

Il s'agit donc de **TESTCOB**. Normalement, tout morceau de **code de production COBOL** comporte **quatre divisions**.

- DIVISION D'IDENTIFICATION
  - Ici, nous identifions le programme COBOL, le nom du programme lui-même. En général, c'est ici que se trouve la documentation du programme.
- DIVISION DE L'ENVIRONNEMENT
  - Ici, nous relierons le programme à des ensembles de données externes.

- DIVISION DES DONNÉES
  - Nous définissons ici les variables locales, exclusives à ce programme uniquement.
- DIVISION DE PROCÉDURE
  - Ici, nous codons toute notre logique.

Comme vous pouvez le voir sur l'image ci-dessous, **TESTCOB** possède en effet quatre divisions.

```

REVEDIT  SYS2.JCLLIB(TESTCOB)                                COLUMNS 00001 00072
COMMAND ==> SCROLL ==> 10
*****ZAP*****AUTOSAVE***** TOP OF DATA *****
000001 //TESTCOB JOB (SETUP),
000002 //      TEST COBOL',
000003 //      CLASS=A,
000004 //      MSGCLASS=A,
000005 //      MSGLEVEL=(1,1)
000006 *****
000007 //
000008 // NAME: SYS2.JCLLIB(TESTCOB)
000009 //
000010 // DESC: TEST COBOL INSTALLATION
000011 //
000012 *****
000013 //HELLOWRLD EXEC COBUCLG
000014 //COB.SYSIN DD *
000015 001 IDENTIFICATION DIVISION.
000016 002 PROGRAM-ID. 'HELLO'.
000017 003 ENVIRONMENT DIVISION.
000018 004 CONFIGURATION SECTION.
000019 005 SOURCE-COMPUTER. IBM-360.
000020 006 OBJECT-COMPUTER. IBM-360.
000021 0065 SPECIAL-NAMES.
000022 0066     CONSOLE IS CNSL.
000023 007 DATA DIVISION.
000024 008 WORKING-STORAGE SECTION.
000025 009 77 HELLO-CONST PIC X(12) VALUE 'HELLO, WORLD'.
000026 075 PROCEDURE DIVISION.
000027 090 000-DISPLAY.
000028 100     DISPLAY HELLO-CONST UPON CNSL.
000029 110     STOP RUN.
000030 //LKED.SYSLIB DD DSN=SYS1.COBLIB,DISP=SHR
000031 //      DD DSN=SYS1.LINKLIB,DISP=SHR
000032 //GO.SYSPRINT DD SYSOUT=A
000033 *****ZAP*****AUTOSAVE***** BOTTOM OF DATA *****

```

Maintenant, qu'est-ce que j'ai mis en surbrillance en rose ? Ce sont des sections. Les sections peuvent être omises mais elles ont choisi de ne pas l'être dans ce cas. À la ligne **000027**, vous pouvez voir **000-DISPLAY**, c'est une routine. Les deux lignes suivantes sont **des instructions/phrases** à l'intérieur d'un **paragraphe**. Elles indiquent que la machine doit **AFFICHER** le **HELLO-CONST** dans **CNSL**. Le **HELLO-CONST** est défini à la ligne **000025** dans la division des données. **CNSL** est un alias pour **CONSOLE** défini dans la section **CONFIGURATION**. Le code parle à peu près de lui-même, n'est-ce pas ? Affichez **HELLO, WORLD** dans la console, puis **ARRÊTEZ** l'exécution.

Maintenant, qu'est-ce que **HELLO-CONST PIC X(12) VALUE 'HELLO, WORLD'** ? Cela signifie que **HELLO-CONST** peut contenir 12 octets. 1 octet équivaut à 1 caractère dans un ordinateur central **EBCDIC**. Vous pouvez voir **X** comme un modèle, en théorie **HELLO-CONST** est **XXXXXXXXXXXX**. Nous remplissons ces X avec 'HELLO, WORLD', X désigne le facteur de répétition, ce qui signifie que X est répété 12 fois.

ALERTE SPOILER : remplacez **MSGCLASS=A** par **MSGCLASS=H** à la ligne 00

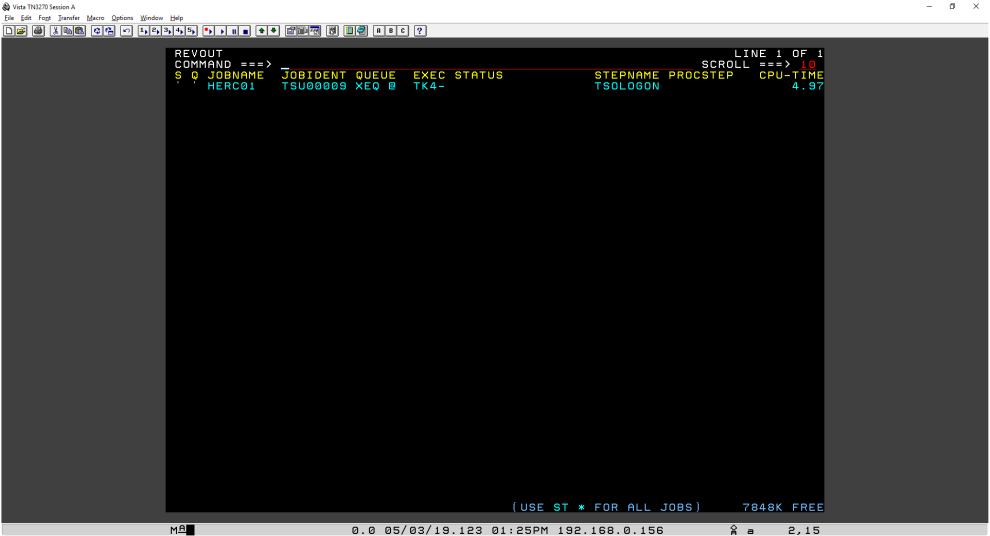


Une fois que vous avez terminé de vérifier ce fichier, entrez la commande principale **save** pour enregistrer vos modifications. Pour exécuter votre programme, entrez la commande principale **submit**.

TEST D'EMPLOI (JOB00002) SOUMIS

Pour vérifier notre sortie, nous devons revenir à la racine **RFE** et accéder à la **OUTLIST (3.8)**. Sélectionnez le travail correct en utilisant la commande de ligne **s**.

## Le blog de codage de Kevin rencontre des difficultés techniques



Ce n'est pas ici ? Où est-il ? Le travail COBOL n'est pas généré vers la liste de sortie ?

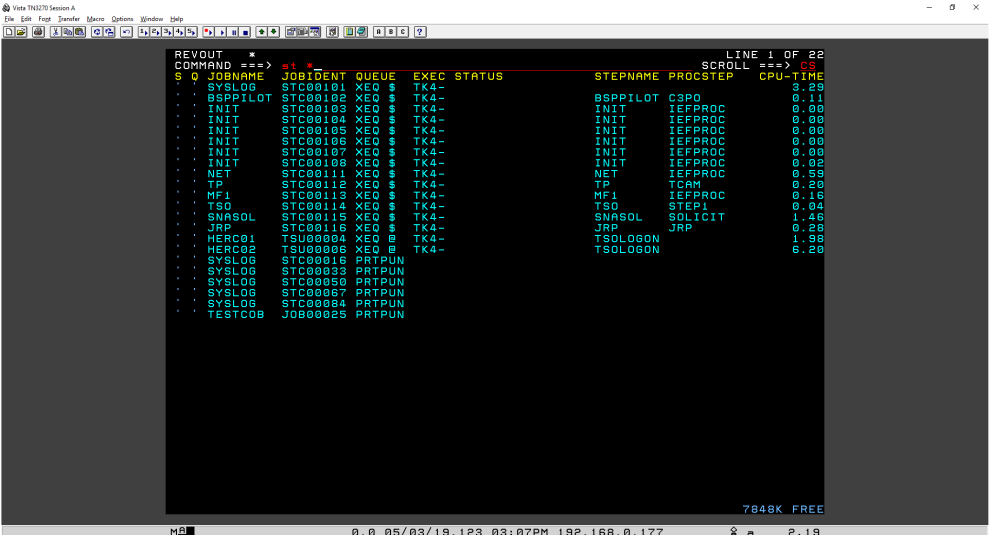
Permettez-moi d'ajouter quelques balises Google pour ceux qui ont le même problème : cobol mainframe MVS 3.8 tk4 pas de sortie de tâche aide sysprint s'il vous plaît google.

```
000001 //TESTCOB JOB (SETUP),
000002 //          'TEST COBOL',
000003 //          CLASS=A,
000004 //          MSGCLASS=H,
000005 //          MSGLEVEL=(1,1)
```

J'ai oublié de changer la **MSGCLASS** en **H**. Cela signifie que vous voulez que la sortie du compilateur soit **conservée** . Maintenant, soumettons-la à nouveau et vérifions la sortie.

TEST D'EMPLOI (JOB00025) SOUMIS

Oui, je sais, 25... J'ai eu du mal. Encore une fois, la sortie du travail était vide, mais après avoir entré la commande principale **ST \***, je peux voir tous les travaux.



TESTCOB est là ! Maintenant je vais sélectionner avec **s** et voir ce qu'il fait.

```

File Edit Find Transfer Macros Window Help
-----
SYS19123.T140755.RA000.HERC02.JOB00025 ----- Line 1 Col 2 81
Command ==> ----- Scroll ==> CS
10 20 30 40 50 60 70 80
-----
J E S 2 J O B L O G
14.05.00 JOB 25 IEF677I WARNING MESSAGE(S) FOR JOB TESTCOB ISSUED
14.05.00 JOB 25 SHASP373 TESTCOB - STARTED - INIT 1 - CLASS A - SYS TK4-
14.05.00 JOB 25 IEF403I TESTCOB - STARTED - TIME=14.05.00
14.05.00 JOB 25 IEC130I SYSPUNCH DD STATEMENT MISSING
14.05.00 JOB 25 IEC130I SYSLIB DD STATEMENT MISSING
14.05.00 JOB 25 IEC130I SYSPUNCH DD STATEMENT MISSING
14.05.00 JOB 25 IEF401I - Stepname Program Retcode
14.05.00 JOB 25 TESTCOB HELOWRLD COB IKFCBL00 RC= 0000
14.05.00 JOB 25 TESTCOB HELOWRLD LKED IEHL RC= 0000
14.05.00 JOB 25 HELOWRLD WORLD
14.05.00 JOB 25 TESTCOB HELOWRLD GO PGM=*.LD RC= 0000
14.05.00 JOB 25 IEF404I TESTCOB - ENDED - TIME=14.05.00
14.05.00 JOB 25 SHASP395 TESTCOB ENDED
1 //TESTCOB JOB (SETUP),
// CLASS='A',
// MSGCLASS=H,
// MSGLEVEL=(1,1),
// USER=HERC02, PASSWORD=
***** GENERATED BY 8DL
***
*** NAME: SYS5.JCLLIB1.TESTCOB)
*** DESC: TEST COBOL INSTALLATION
***
*****
2 //HELOWRLD EXEC COBUCLG
3 XXCOCUCLG PROC SOUT='*'
4 XXCOC EXEC PGM=IKFCBL00,
XX PARM='LOAD,SUPMAP,SIZE=040K,BUF=1024K'
5 XXSYSPRINT DD SYSOUT=&SOUT
6 XXSYSDA DD UNIT=SYSDA,SPACE=(400,1700,100)
7 XXSYSDA DD UNIT=SYSDA,SPACE=(400,1700,100)
8 XXSYSDA DD UNIT=SYSDA,SPACE=(400,1700,100)
9 XXSYSDA DD UNIT=SYSDA,SPACE=(400,1700,100)
10 XXSYSLIN DD DSNNAME=&LOADSET,DISP=(MOD,PASS),UNIT=SYSDA,
XX SPACE=(80,1700,100)
7820K FREE
M5 0.05/03/19.123 03:08PM 192.168.0.177 8 2.15

```

Et voilà ! **BONJOUR LE MONDE** . Nous l'avons fait !

Prochainement : notre propre mix personnalisé de **JCL** et **COBOL** !

## Rencontrez-vous des difficultés techniques ?

Raconte-moi tout !

**Tagué:** 3.8j, cobol, Hercule, JCL, unité centrale, MVS, MVS3.8, programmation, Clé en main

Publié par kevindurant

[Voir tous les articles de kevindurant](#)



< MOM Pt. 002 : Écran de connexion NETSOL personnalisé

MOM Pt. 004 : Écriture simple en COBOL sur MVS3.8 tk4- >

## 5 réflexions sur « MOM Pt. 003 : Découvrir COBOL – Compilation »



**Carsten Spräner**

12 février 2023 à 21h30

Merci pour ce blog ! Ils m'ont ouvert un tout nouveau monde. Un monde que je ne connaissais que par mes collègues qui parlaient de files d'attente de tâches, de JCL et de jeux de données. En tant que programmeur Java, c'est vraiment bien de savoir de quoi ils parlent.

↳ Répondre



**Jean Dupont**

9 mars 2021 à 02h36

Le script de test génère plusieurs erreurs. Nous obtenons :

03.11.20 JOB 31 IEF677I MESSAGE(S) D'AVERTISSEMENT POUR LE TRAVAIL TESTCOB ÉMIS

03.11.20 JOB 31 \$HASP373 TESTCOB DÉMARRÉ - INIT 1 - CLASSE A - SYS TK4-  
03.11.20 JOB 31 IEF403I TESTCOB - DÉMARRÉ - HEURE=03.11.20  
03.11.20 JOB 31 IEC130I DÉCLARATION DD SYSPUNCH MANQUANTE  
03.11.20 JOB 31 IEC130I DÉCLARATION DD SYSLIB MANQUANTE  
03.11.20 JOB 31 IEC130I DÉCLARATION DD SYSPUNCH MANQUANTE  
03.11.20 JOB 31 IEFACTRT - Nom d'étape Procstep Code de retour du programme  
03.11.20 JOB 31 TESTCOB HELOWRLD COB IKFCBL00 RC= 0000  
03.11.20 JOB 31 TESTCOB HELOWRLD LKED IEWL RC= 0000  
03.11.20 JOB 31 +HELLO, WORLD  
03.11.20 JOB 31 TESTCOB HELOWRLD GO PGM=\*.DD RC= 0000  
03.11.20 JOB 31 IEF404I TESTCOB – TERMINÉ – HEURE=03.11.20  
03.11.20 JOB 31 \$HASP395 TESTCOB TERMINÉ

[↳ Répondre](#)**Georges**

29 octobre 2019 à 03h29

Salut Kevin,

Dans le chapitre 4 partie 3 j'ai eu le même problème, mais Bob L du mainframe slack m'a donné la réponse : Dans SDSF, essayez : préfixe \* puis : propriétaire Z##### le nom du propriétaire de l'œuvre. Cela devrait fonctionner dans votre MOM aussi !

À vous de me le dire !

Georges

[↳ Répondre](#)**kévindurant**

29 octobre 2019 à 07:33

Bonjour Georges,

Intéressant ! Merci pour le partage ! Je vais essayer ça, j'ai eu quelques difficultés lors des derniers défis MTM 19 dont j'ai parlé sur mon blog. Je n'ai pas vu mes résultats mais je savais ce que je faisais donc je n'en avais pas vraiment besoin. Je suppose que vous pourriez dire que j'étais paresseux, heheh.

Santé!

– Kevin

[↳ Répondre](#)**Marco**

6 mai 2019 à 14h23

Les choses seraient plus simples si vous changiez le nom jcl de TESTCOB en quelque chose comme HERC01C, où HERC01 est le profil utilisateur que vous utilisez

[↳ Répondre](#)

## Laisser un commentaire

Votre adresse email ne sera pas publiée. Nom, email et site web non requis.

Commentaire

Nom

E-mail

Site web

☐

Enregistrer mon nom, mon e-mail et mon site web dans le navigateur pour mon prochain commentaire.

Poster un commentaire

### Articles récents

**IBM MTM 2020 : mise à jour du blog**

**Maîtrisez le mainframe à l'aide de Zowe CLI et VSCode**

**IBM MTM 2019 : le mot de la fin**

**IBM MTM 2019 : Troisième partie – Défi n°15**

**IBM MTM 2019 : Troisième partie – Défi n°14**

### Recevez des mises à jour par e-mail !

E-mail\*

Soumettre

### Recherche

Recherche ...

### Commentaires récents

**Britt Briseno** sur **IBM MTM 2019 : Deuxième partie – Défi n°13**

**Howard Lewin** sur **IBM MTM 2019 : première partie – Défi n° 01**

**Charlie Isabel** sur **IBM MTM 2018 : Deuxième partie – Challenge #05**

**Zane Cadman** sur **IBM MTM 2018 : Troisième partie – Défi n°14**

**Jake Wootton** sur **IBM MTM 2019 : le mot de la fin**

## Archives

[Septembre 2020](#)

[Avril 2020](#)

[Janvier 2020](#)

[Décembre 2019](#)

[Novembre 2019](#)

[Octobre 2019](#)

[Septembre 2019](#)

[Juillet 2019](#)

[Mai 2019](#)

[Mars 2019](#)

[Janvier 2019](#)

[Décembre 2018](#)

[Novembre 2018](#)

[Octobre 2018](#)

[Septembre 2018](#)

## Catégories

[IBM Maîtrisez le mainframe 2018](#)

[IBM Maîtrisez le mainframe 2019](#)

[IBM Master The Mainframe 2020](#)

[Unité centrale](#)

[Mon propre mainframe](#)

[Zoé](#)

## Méta

[Se connecter](#)

[Entrées RSS](#)

[Commentaires RSS](#)

[WordPress.org](#)