

EXEMPLE Cobol 1

-----78-----1
2

IDENTIFICATION DIVISION.
PROGRAM-ID. CHANGE.
AUTHOR. GR.

```
*-----*
*           P R O G R A M M E           *
*   D E V I S E S E T R A N G E R E S   *
*   COURS DU 29 OCTOBRE 1997           *
*-----*
```

ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. MVS.
OBJECT-COMPUTER. MVS.

INPUT-OUTPUT SECTION.
FILE-CONTROL.

SELECT IMPRESSION ASSIGN TO IMPR.

DATA DIVISION.

FILE SECTION.

FD IMPRESSION.

01 PAPIER.
05 LIGNE PIC X(80).

WORKING-STORAGE SECTION.

```
*-----*
* DEFINITION DES VARIABLES *
*-----*
```

77 TEST-EUROPE PIC X(3).
88 EUROPE VALUE "DEM" "ATS" "BEF" "DKK" "ESP" "FIM" "GBP"
"GRD" "IEP" "ITL" "NLG" "PTE" "SEK".

01 ENTRER PIC X VALUE SPACE.
01 INDICE PIC 99 VALUE ZERO.
01 IND PIC 99 VALUE ZERO.
01 IMPRESS PIC X VALUE "N".

01 MONTANT-DEVISE PIC X(13) VALUE ZERO.

01 VERSEMENT REDEFINES MONTANT-DEVISE.
05 V-DEVISE PIC X(3).
05 FILLER PIC X.
05 V-MONTANT PIC 9(5)V9(3).

01 TABLE-DEVISE.
05 FILLER PIC X(32) VALUE "ALLEMAGNE DEM 003.3400".
05 FILLER PIC X(32) VALUE "AUSTRALIE AUD 004.0800".
05 FILLER PIC X(32) VALUE "AUTRICHE ATS 047.4000".
05 FILLER PIC X(32) VALUE "BELGIQUE BEF 016.2100".
05 FILLER PIC X(32) VALUE "CANADA CAD 004.1400".
05 FILLER PIC X(32) VALUE "DANEMARK DKK 087.0000".
05 FILLER PIC X(32) VALUE "ESPAGNE ESP 003.9500".

05 FILLER PIC X(32) VALUE "FINLANDE FIM 001.0830".
05 FILLER PIC X(32) VALUE "ANGLETERRE GBP 009.7050".
05 FILLER PIC X(32) VALUE "GRECE GRD 002.1000".
05 FILLER PIC X(32) VALUE "REPUBLIQUE TCHEQUE CZK 000.1800".
05 FILLER PIC X(32) VALUE "IRLANDE IEP 008.6400".
05 FILLER PIC X(32) VALUE "ITALIE ITL 034.0000".
05 FILLER PIC X(32) VALUE "JAPON JPY 004.8050".
05 FILLER PIC X(32) VALUE "NORVEGE NOK 081.0000".
05 FILLER PIC X(32) VALUE "PAYS BAS NLG 002.9600".
05 FILLER PIC X(32) VALUE "PORTUGAL PTE 003.2500".
05 FILLER PIC X(32) VALUE "SUEDE SEK 076.5000".
05 FILLER PIC X(32) VALUE "SUISSE CHF 004.0700".
05 FILLER PIC X(32) VALUE "ETATS UNIS USD 005.8000".
05 FILLER PIC X(32) VALUE "HONGRIE HUF 000.0315".
05 FILLER PIC X(32) VALUE "SLOVAQUIE SKK 000.1900".

01 TABLE-COURS REDEFINES TABLE-DEVISE.

05 COURS OCCURS 22.
10 TBL-PAYS PIC X(19).
10 TBL-DEVISE PIC X(3).
10 FILLER PIC XX.
10 TBL-COURS PIC 9(3).9(4).

01 CALCUL-COURS PIC 9(3)V9(4) VALUE ZERO.

01 CALCUL.

05 BRUT PIC 9(5)V9(2) VALUE ZERO.
05 COMMISSION PIC 9(3)V99 VALUE ZERO.
05 TVA PIC 9(3)V99 VALUE ZERO.
05 NET PIC 9(5)V99 VALUE ZERO.

01 EDITION1.

05 ED-MONTANT PIC Z(5).99 VALUE ZERO.
05 ED-COURS PIC Z(2)9.9(4) VALUE ZERO.
05 ED-BRUT PIC Z(5).99 VALUE ZERO.
05 ED-COMMISSION PIC Z(2)9.99 VALUE ZERO.
05 ED-TVA PIC Z(2)9.99 VALUE ZERO.
05 ED-NET PIC Z(5).99 VALUE ZERO.

PROCEDURE DIVISION.

```
*-----*
* D E B U T D U P R O G R A M M E *
*-----*
```

DEBUT.

PERFORM INITIALISATION.
PERFORM AFFICHAGE VARYING INDICE FROM 1 BY 2 UNTIL
INDICE > 22.
DISPLAY SPACE UPON CONSOLE.
PERFORM SAISIE UNTIL ENTRER = "F".
STOP RUN.

```
*-----*
* BLOCS APPELES *
*-----*
```

```
* I N I T I A L I S A T I O N
*
```

INITIALISATION.

```
* MOVE SPACE TO MONTANT-DEVISE.
* MOVE 0 TO BRUT COMMISSION TVA NET.
```

```

*
* A F F I C H A G E
*
AFFICHAGE.
  DISPLAY TBL-PAYS(INDICE) UPON CONSOLE NO ADVANCING.
  DISPLAY TBL-DEVISE(INDICE) UPON CONSOLE NO ADVANCING.
  DISPLAY "      " UPON CONSOLE NO ADVANCING.
  DISPLAY TBL-PAYS(INDICE + 1) UPON CONSOLE NO ADVANCING.
  DISPLAY TBL-DEVISE(INDICE + 1) UPON CONSOLE.

*
* S A I S I E
*
SAISIE.
  DISPLAY "Entrez devise et somme (dev 99999.99)" UPON CONSOLE.
  DISPLAY "dev étant la devise sur 3 caractères" UPON CONSOLE.
  DISPLAY "*** Format IMPOSE !!! ***" UPON CONSOLE.
  ACCEPT MONTANT-DEVISE FROM CONSOLE.
  MOVE V-DEVISE TO TEST-EUROPE.
  PERFORM TRAIT.
  DISPLAY "SUITE (F POUR FIN)" UPON CONSOLE NO ADVANCING.
  ACCEPT ENTRER FROM CONSOLE.

*
* T R A I T E M E N T
*
TRAIT.
  MOVE 0 TO INDICE.
  PERFORM RECHERCHE VARYING INDICE FROM 1 BY 1 UNTIL
  INDICE > 22.

*
* R E C H E R C H E   D U   C O U R S
*
RECHERCHE.
  IF TBL-DEVISE(INDICE) = V-DEVISE
    MOVE INDICE TO IND
    MOVE 22 TO INDICE
    PERFORM EDITION
  ELSE
    IF INDICE = 22
      DISPLAY V-DEVISE "CETTE DEVISE N'EXISTE PAS"
      UPON CONSOLE.

*
* E D I T I O N   A   L   '   E C R A N
*
EDITION.
* AFFICHAGE LIGNE VERSEMENT
  DISPLAY "      " UPON CONSOLE
  NO ADVANCING.
  MOVE V-MONTANT TO ED-MONTANT.
  MOVE TBL-COURS(IND) TO ED-COURS.
  DISPLAY "VERSEMENT :      " ED-MONTANT " " V-DEVISE
  UPON CONSOLE.
  DISPLAY "COURS DU " V-DEVISE " : " ED-COURS
  UPON CONSOLE.
* CALCUL BRUT
  MOVE TBL-COURS(IND) TO CALCUL-COURS.
  COMPUTE BRUT = V-MONTANT * CALCUL-COURS.
  MOVE BRUT TO ED-BRUT.
* AFFICHAGE LIGNE BRUT
  DISPLAY "      " UPON CONSOLE

```

```

NO ADVANCING.
  DISPLAY "CHANGE BRUT :      " ED-BRUT " F" UPON CONSOLE.
* CALCUL COMMISSION
  IF EUROPE
    COMPUTE COMMISSION = BRUT * 0.04
  ELSE
    COMPUTE COMMISSION = BRUT * 0.06.
  MOVE COMMISSION TO ED-COMMISSION.
* AFFICHAGE LIGNE COMMISSION
  DISPLAY "FRAIS DE CHANGE :      " "COMMISSION HT " " - "
  UPON CONSOLE NO ADVANCING.
  DISPLAY ED-COMMISSION " F" UPON CONSOLE.
* CALCUL TVA
  COMPUTE TVA = COMMISSION * 0.206.
  MOVE TVA TO ED-TVA.
* AFFICHAGE LIGNE TVA
  DISPLAY "      " UPON CONSOLE
  NO ADVANCING.
  DISPLAY "TVA (20,6%) -      " ED-TVA " F" UPON CONSOLE.
* CALCUL NET
  COMPUTE NET = BRUT - COMMISSION - TVA.
  MOVE NET TO ED-NET.
* AFFICHAGE LIGNE TVA
  DISPLAY "      " UPON CONSOLE
  NO ADVANCING.
  DISPLAY "CHANGE NET      " ED-NET " F" UPON CONSOLE.

  DISPLAY "VOULEZ-VOUS UNE SORTIE A L'IMPRIMANTE (O/N)"
  UPON CONSOLE NO ADVANCING.
  ACCEPT IMPRESS FROM CONSOLE.
  IF IMPRESS = "O"
    PERFORM IMPRESSION-FICHER.

*
* E D I T I O N   A   L   '   I M P R I M A N T E
*
IMPRESSION-FICHER.
  OPEN EXTEND IMPRESSION.
  STRING "      VERSEMENT :      " ED-MONTANT
  " " V-DEVISE DELIMITED BY "*" INTO LIGNE.
  WRITE PAPIER.
  MOVE SPACES TO LIGNE.
  STRING "COURS DU " V-DEVISE " : " ED-COURS
  DELIMITED BY "*" INTO LIGNE.
  WRITE PAPIER.
  MOVE SPACES TO LIGNE.
  STRING "      CHANGE BRUT :      " ED-BRUT
  " F" DELIMITED BY "*" INTO LIGNE.
  WRITE PAPIER.
  MOVE SPACES TO LIGNE.
  STRING "FRAIS DE CHANGE :      COMMISSION HT - "
  ED-COMMISSION " F" DELIMITED BY "*" INTO LIGNE.
  WRITE PAPIER.
  MOVE SPACES TO LIGNE.
  STRING "      TVA (20,6%) -      " ED-TVA
  " F" DELIMITED BY "*" INTO LIGNE.
  WRITE PAPIER.
  MOVE SPACES TO LIGNE.
  STRING "      CHANGE NET      " ED-NET
  " F" DELIMITED BY "*" INTO LIGNE.
  WRITE PAPIER.
  MOVE SPACES TO LIGNE.
  MOVE "N" TO IMPRESS.

```

EXEMPLE Cobol 2

IDENTIFICATION DIVISION.
PROGRAM-ID. PUBLI.

AUTHOR. GR.

***-----

ENVIRONMENT DIVISION.

CONFIGURATION SECTION.
SOURCE-COMPUTER. MWB.
OBJECT-COMPUTER. MWB.
INPUT-OUTPUT SECTION.
FILE-CONTROL.

SELECT BASE ASSIGN TO BASABON
ORGANIZATION RECORD SEQUENTIAL
ACCESS SEQUENTIAL
FILE STATUS IS BS.

SELECT MOUV ASSIGN TO MAJABON
ORGANIZATION RECORD SEQUENTIAL
ACCESS SEQUENTIAL
FILE STATUS IS MS.

SELECT NOUBASE ASSIGN TO NEWABON
ORGANIZATION RECORD SEQUENTIAL
ACCESS SEQUENTIAL
FILE STATUS IS NS.

SELECT ERREUR ASSIGN TO ERRABON
ORGANIZATION LINE SEQUENTIAL
ACCESS SEQUENTIAL
FILE STATUS IS ES.

***-----

DATA DIVISION.

FILE SECTION.

*-----

FD BASE LABEL RECORD STANDARD.

01 B-ENR.
* Code-adresse
05 B-CD-ADR PIC 9(6).
* Cle de controle
05 B-CLE PIC X.
* Nom-Prenom
05 B-NM-PRNM PIC X(25).
* Adresse
05 B-ADR PIC X(40).
* Code Postal
05 B-CD-PSTL PIC 9(5).
* Ville
05 B-VL PIC X(25).

*-----

FD MOUV LABEL RECORD STANDARD.

01 M-ENR.
05 M-CD-OPR PIC X.
05 M-DONNEES.
10 M-CD-ADR PIC 9(6).
10 M-CD-ADR REDEFINES M-CD-ADR PIC X(6).
10 M-CLE PIC X.
10 M-NM-PRNM PIC X(25).
10 M-ADR PIC X(40).
10 M-CD-PSTL PIC 9(5).
10 M-VL PIC X(25).

*-----

FD NOUBASE LABEL RECORD STANDARD.
01 N-ENR PIC X(102).

*-----

FD ERREUR LABEL RECORD STANDARD.
01 E-ENR PIC X(103).

*-----

WORKING-STORAGE SECTION.
* Code etat du fichier BASE
77 BS PIC XX.
* Code etat du fichier MOUV
77 MS PIC XX.
* Code etat du fichier NOUBASE
77 NS PIC XX.
* Code etat du fichier ERREUR
77 ES PIC XX.
* Zone de travail
77 QUOTIENT PIC 9(6) COMP.
* Rang de la lettre de controle
77 RG-LTR PIC 99 COMP.
88 CORRECTE VALUE 1 THRU 23.
* Rang du message d'erreur
77 RG-ERR PIC 9.

* Definition des cles de controle
01 LETTRES PIC X(23) VALUE "ABCDEFGHJKLMNPQRSTUVWXYZ".
01 TABCLE REDEFINES LETTRES.
05 CLE PIC X OCCURS 23.

* Definition des messages d'erreur
01 LST-MSG.
05 FILLER PIC X(30) VALUE "(1) CLE ERRONEE".
05 FILLER PIC X(30) VALUE SPACES.
05 FILLER PIC X(30) VALUE SPACES.
05 FILLER PIC X(30) VALUE "(4) CODE OPER. ERRONEE".
05 FILLER PIC X(30) VALUE "(5) DONNEE INCORRECTE".
05 FILLER PIC X(30) VALUE SPACES.
05 FILLER PIC X(30) VALUE SPACES.
05 FILLER PIC X(30) VALUE SPACES.
01 TABERR REDEFINES LST-MSG.
05 MSGERR PIC X(30) OCCURS 8.

***-----

PROCEDURE DIVISION.
DECLARATIVES.

LECT SECTION. USE AFTER STANDARD ERROR PROCEDURE ON INPUT.
L0. IF MS = 10 OR BS = 10 NEXT SENTENCE
ELSE

```

DISPLAY "Erreur d'entrée-sortie en lecture" UPON CONSOLE
DISPLAY "Code d'etat des fichiers:" UPON CONSOLE
DISPLAY "BASABON = " BS "      MAJABON = " MS UPON CONSOLE
CLOSE MOUV BASE NOUBASE ERREUR.
STOP RUN.

ECR SECTION. USE AFTER STANDARD ERROR PROCEDURE ON OUTPUT.
E0.
  DISPLAY "Erreur d'entrée-sortie en ecriture" UPON CONSOLE
  DISPLAY "Code d'etat des fichiers:" UPON CONSOLE
  DISPLAY "NOUBASE = " NS "      ERREUR = " ES UPON CONSOLE
  CLOSE MOUV BASE NOUBASE ERREUR.
  STOP RUN.

END DECLARATIVES.

PUBLI SECTION.

* Initialisation du traitement
INIT.
  OPEN      INPUT  BASE MOUV
           OUTPUT NOUBASE ERREUR.
  PERFORM LIRBAS.
  PERFORM LIRMAJ.

* Traitement majeur
CORPS.
  PERFORM PRINCIP UNTIL BS NOT = ZERO OR MS NOT = ZERO.
  PERFORM FINBAS UNTIL BS NOT = ZERO.
  PERFORM FINMAJ UNTIL MS NOT = ZERO.

* Finalisation du traitement
FIN.
  CLOSE BASE MOUV NOUBASE ERREUR.
  DISPLAY "FIN DE PUBLI" UPON CONSOLE.
  STOP RUN.

*-----
SPECIALE SECTION.

  * Repetitive principale
PRINCIP. PERFORM CALCLE. IF CORRECTE PERFORM TRAIT
        ELSE PERFORM ERR PERFORM LIRMAJ.

  * Traitement final de BASE
FINBAS. PERFORM COPIE PERFORM LIRBAS.

  * Traitement final de MOUV
FINMAJ. IF M-CD-OPR = "C" PERFORM AJOU
        ELSE MOVE 4 TO RG-ERR PERFORM ERR.
        PERFORM LIRMAJ.

LIRBAS. READ BASE.

LIRMAJ. READ MOUV.

  * Validation de la cle
CALCLE. IF M-CD-ADRX NOT NUMERIC
        MOVE ZERO TO RG-LTR
        MOVE 5 TO RG-ERR
        ELSE
          DIVIDE 23 INTO M-CD-ADR      GIVING QUOTIENT
                                REMAINDER RG-LTR
          SUBTRACT RG-LTR FROM 23 GIVING RG-LTR

```

```

IF CLE(RG-LTR) NOT = M-CLE
  MOVE ZERO TO RG-LTR
  MOVE 1 TO RG-ERR.

  * Deroulement de BASE
TRAIT.  IF M-CD-ADR > B-CD-ADR PERFORM COPIE PERFORM LIRBAS
        ELSE PERFORM NONSUP.

  * Ecriture des erreurs
ERR.    WRITE E-ENR FROM M-ENR. WRITE E-ENR FROM MSGERR(RG-ERR).

  * Ecriture de NOUBASE
COPIE.  WRITE N-ENR FROM B-ENR.

  * Ajout d'un mouvement en creation
AJOU.
*----- Il faudrait controler la validite des rubriques!!
        WRITE N-ENR FROM M-DONNEES.

NONSUP. IF M-CD-ADR = B-CD-ADR
        PERFORM EGAL
        ELSE IF M-CD-OPR = "C"
          PERFORM AJOU
          ELSE MOVE 4 TO RG-ERR PERFORM ERR.

        PERFORM LIRMAJ.

  * Code-adresse existant dans BASE
EGAL.   IF M-CD-OPR = "S"
        NEXT SENTENCE
        ELSE IF M-CD-OPR = "M"
          PERFORM MODIF
          PERFORM COPIE
          ELSE PERFORM COPIE
          MOVE 4 TO RG-ERR PERFORM ERR.

        PERFORM LIRBAS.

  * Chargement des champs a modifier
MODIF.  IF M-NM-PRNM NOT = SPACES MOVE M-NM-PRNM TO B-NM-PRNM.
        IF M-ADR NOT = SPACES MOVE M-ADR TO B-ADR.
        IF M-CD-PSTL NOT = SPACES MOVE M-CD-PSTL TO B-CD-PSTL.
        IF M-VL NOT = SPACES MOVE M-VL TO B-VL.

```