Programme de service pour formater un nombre

- Les articles -

Date de mise en ligne : mardi 6 novembre 2007

Environnement iSeries

Formater facilement les nombres en texte.

Introduction :

SGFORMAT est un programme de service qui contient une procédure exportée "FormatS" permettant de formater un nombre.

Grégory Jarrige à écrit une procédure similaire essentiellement basée sur les BIF %EDITC et %EDITW (Voir article sur EditC).

FormatS offre une approche un peu différente, il est possible par exemple de spécifier la longueur attendue ou encore de changer le séparateur décimal, d'ignorer ou conserver les zéros non significatifs, d'assigner un séparateur de millier...

Prenons le chiffre -12,45 (choix complètement arbitraire), selon les besoins le programmeur peut vouloir des résultats différents comme :

- -12,45 (formatage classique)
- 12,45 (pas de prise en compte du signe)
- -12.45 (le point comme caractère décimal)
- 00012,450 (formater en longueur 8 avec 3 décimales et 0 conservés)
- $$$12,45$ (formater en longueur 8 avec 3 décimales et 0 remplacés par $)
- $$#12.45#$ (comme précédent et # comme séparateur de millier)

Tous ces "formatages" (et bien d'autres encore) sont possible avec la procédure FormatS (voir chapitre exemple d'appel pour le détail). Pour cela j'ai utilisé la technique des paramètre facultatifs qui permet, dans la plupart des cas, un codage très simple du style "FormatS(W_NB)" , car le plus souvent c'est ainsi que je souhaite formater mes nombres. Seul le 1er paramètre (qui transmet le nombre) est obligatoire, l'utilisation des autres paramètres permet des formatages plus complexe s quand c'est nécessaire..

Description Technique

FormatS reçoit comme 1er paramètre un nombre et retourne une chaine de 35 caractères contenant ce nombre formaté. La procédure peut recevoir jusqu'à 7 paramètres, mais seul le 1er est obligatoire.

- Description des paramètres :

— 1=>P_Val_Ori (30P9) est le chiffre à formater. Ce 1er paramètre est passer par valeur, ce qui permet de passer une valeur littérale.
— 2=>P_Lng (2P0) est la longueur totale attendue (maxi 30). Ce 2ème paramètre est facultatif et n'est utile que si vous souhaitez conservez ou substituer les zéros non significatifs.
— 3=>P_Dec (1P0) nombre de décimale attendue.
— 4=>P_Zero (1A) ce paramètre facultatif (par défaut on considère que les 0 non significatifs sont supprimés) peut prendre 4 valeurs (voir le prototype). On mettra *blank dans P_Zero si on veut conserver les 0.
— 5=>P_Subst(1A) ce paramètre facultatif (par défaut = *blank) permet de remplacer les zéros non significatifs par un caractère quelconque. Ce paramètre sera sans effet si P_Zero = *BLANK.

— 6=>P_Moins(1N) ce paramètre facultatif (par défaut = *on) permet de prendre en compte ou pas, le signe -. Pour ne pas afficher de moins quand le nombre est négatif, on affectera *off à P_Moins.
— 7=>P_Sep_Dec (1A) ce paramètre facultatif (par défaut = ",") permet de spécifier un séparateur décimal différent de ",". Si vous affecter *blank à ce paramètre il n'y aura pas de séparateur décimal donc, veillez à alimenter par ailleurs P_Zero avec une valeur <> '-' sinon 12,45 (définit en 8 dont 3) pourrait devenir '1245' avec les P_Zéro = *blank on aurait '00001245'.
— 8=>P_Sep_Mil (1A) ce paramètre facultatif (par défaut = "0") permet de spécifier un séparateur de millier.La valeur '0' ne correspond à aucun séparateur.

Exemple d'appel :

Voyons maintenant comment utiliser ce programme de service. Ce programme devra être lié à la compilation au programme de service SGFORMAT.

```
H DECEDIT('0,') DATEDIT(*YMD)      /COPY PROTOTYPE,SGFORMAT     D W_Orig         S            30P 9       D W_A35          S            35A      /Free       *Inlr  =  *on ;

W_Orig = -12,45 ;          //  -12,45 (formatage classique)        W_A35  = FORMATS(W_Orig) ;          //  12,45 (pas de prise en compte du signe)        W_A35  = FORMATS(W_Orig:4:2 :'-'
:*OFF) ;          //  -12.45 (le point comme caractère décimal)        W_A35  = FORMATS(W_Orig:4:2 :'-' :*ON :' ' :'.') ;          // -00012,450 (formater en longueur 8 avec 3 décimales et 0
conservés)        W_A35  = FORMATS(W_Orig:8:3 :' ') ;          // -$$$12,45$ (formater en longueur 8 avec 3 décimales et 0 substitués)        W_A35  = FORMATS(W_Orig:8:3 :'-' :*ON :'$') ;
// -$$#$12.45$#$ (le point comme caractère décimal et # comme séparateur de millier)        W_A35  = FORMATS(W_Orig:9:4 :'-' :*ON :'$' :'.' :'#') ;


/End-Free
```

Prototype

```
/IF DEFINED(SGFORMAT)      /EOF      /ENDIF      /DEFINE SGFORMAT      *----------------------------------------------------      * Procedure name : FormatS      * Purpose :       Formater un
nombre en chaine      * Returns :       Chaine de 40 caractères 1     * Parameter :      P_Val_Ori => Valeur d'origine 2     * Parameter :      P_Lng => Longeur totale attendue 3     * Parameter :
 P_Dec => Nombre de décimales 4     * Parameter :      P_Zero => "<"=>Supprime zéros de gauche      *                          ">"=>Supprime zéros droite      *                          "-"=>Supprime tous
les zéros      *                          " "=>Ne Supprime pas les zéros 5     * Parameter :      P_subst => caractère de substitution des zéros      *                          sans effet si P_Zero = *blank 6     *
Parameter :      P_Moins => Gestion du signe      *                          *OFF=>pas de signe      *                          *ON=>signe - devant 7     * Parameter :      P_Sep_Dec => Séparateur
décimale 8     * Parameter :      P_Sep_Mil => Séparateur millier      *                          => '0' pas de séparateur      *                          => sinon le caractère transmis      * Exemple=> W_A40 =
FORMATS(W_Orig3:8:2 :'-' :*ON :' ' :'.') ;      *      *----------------------------------------------------      D FormatS         PR            40A 1     D  P_Val_Ori                  30P 9 VALUE 2     D  P_Lng
       2P 0 VALUE OPTIONS(*NOPASS) 3     D  P_Dec                    1P 0 VALUE OPTIONS(*NOPASS) 4     D  P_Zero                   1A   VALUE OPTIONS(*NOPASS)      D*
        Valeur par défaut '-' 5     D  P_Moins                   N   VALUE OPTIONS(*NOPASS)      D*                          Valeur par défaut *on 6     D  P_Subst                  1A   VALUE
OPTIONS(*NOPASS)      D*                          Valeur par défaut *blank 7     D  P_Sep_Dec                1A   VALUE OPTIONS(*NOPASS)      D*                          Valeur par défaut ','
8     D  P_Sep_Mil                1A   VALUE OPTIONS(*NOPASS)      D*                          Valeur par défaut 0
```

Téléchargement :

Codes sources

Post-scriptum :

Le fichier joint contient le code source, le prototype et le source du liage.