KKEX

KICKS Spreadsheet

General Information

KKEX is a very simple spreadsheet program. It is based on some code fragments I found. I have added numerous improvements, a complete math parser, saving workspaces numerous functions and loading workspaces.

After every cell entry, the spreadsheet recomputes **ALL** cells.

The maximum size of the spreadsheet is 30(rows) and 26(columns).

Rows are labeled with numbers, starting from 1 to 30, and columns are labeled with letters, starting from A to Z. Cells should be referenced by a letter number and row number, with no space or symbol between; for example, E9, B18. Lowercase and uppercase letters to reference cells are allowed and will be recognized.

KKEX using both integer and double (floating) numbers. They can be intermixed. The output from functions is always in double format.

Any questions, bug reports or comments, send to:
tchandler48@gmail.com

Input

To enter a formula into the spreadsheet, you must enter a CELL location in the the CELL -> field, then press the **TAB** key, then in the VALUE -> field, enter the data you wish to store in the cell location.

Commands

The following are commands that KKEX processes. These commands are preformed via PF Keys.

PF1 **EXIT**

Exit the KKEX spreadsheet. All changes from the LAST save (PF3) will be lost. Save (PF3) your data before exiting the program.

PF2 **LOAD**

Displays a BMS MAP that prompts for the workspace name to be loaded. The workspace will be loaded and executed.

PF3 **SAVE**

Prompts for the name of the workspace file to save the current workspace in.

PF4 **NEW**

Clears the current workspace. If you have not performed a PF3 (save), the any changes since the last save will be lost.

PF5 **CELLS**

This is the default view for the KKEX program. Cells may be updated on this view. After each input, the COMPLETE workspace is recomputed. You may navigate to other views or move the current view around based what PF KEY is pressed.

PF6 **PROG**

This PFK KEY displays the program code for the current workspace. All the functions, math variables, and text variables are displayed in the cells. You MUST press PF5 to return back to the processing mode of the spreadsheet.

PF7 **HOME**

Returns the view port to the stating location. row = 0, column = 1 (Default Home position).

PF8 **BOT**

Positions the spreadsheet to the bottom row. Does NOT change the column value;

PF9 **TOP**

Return to view to row 0,the top of sheet. The column value is NOT changed. You must use *LEFT or *RIGHT to shift the view over.

PF10 **LEFT**

Moves the current view LEFT 4 columns.

PF11 **RIGHT**

Move the current view RIGHT 4 columns.

PF12 **VIEW**

Sets the view port to be displayed on the screen. You will be prompted for the ROW and then the COL entry.

ROW can be 1 to 30.
COL can be 1 to 26.

Math Processing

A math parser written by Dr. Jack Crenshaw has been integrated into **KKEX**. Operations supported are +, -, *, and /.

Mixed mode is supported.

```
     =a1+b1 ...etc
or
     =6+9  or =6+a1  or =b2+7
```

Built-in Functions

You can include functions in the spreadsheet. Functions are of the form =AVG(A2,A5), =SUM(C1,C15). Functions **MUST** start with the = (equal) character.

AVG

AVG will return the average value from the cells inside the (...).
Ex: =AVG(B1,C1);

DATE

DATE will display in the cell, the current date. This cell CAN NOT be used in any math activity. If you need math ability, then use MTH, DAY or YR.
Ex: =DATE
01/20/2021

DAY

Will return the current day as an integer.
Ex: =DAY

HOUR

Will return the current hour as an integer.
Ex: =HOUR

MIN

Will return the current minute as an integer.
Ex: =MIN

MTH

Will return the current month as an integer.
Ex: =YR

POW

Raises a number to the indicate power.
Ex: =POW(a8,2)
a8 = 5
This raises a8 (5) to the 2nd power
giviing the result of 25.

RNG

RNG will return the difference between the largest and smallest value in the input.
Ex: =RNG(C1,C3)

SEC

Will return the current second as an integer.

Ex: =SEC

SQRT

Will return the square root of a cell. The cell to be used MUST be entered as a double (Ex: 4.0, 9.0, 15.33)
Ex: B5 set value 9.0
=SQRT(B5)
Value returned is 3.0

SUM

SUM will return the cumulative total. The input must be a horizontal or vertical 1D range of cells, described by the end points. The formula will compute a result using only values within the given range of cells; it should ignore empty cells and those containing text or another formula. If there are no values within the given range of cells, the result should default to zero.
Ex: =SUM(C1,C3)

TIME

Will return the current time in a string.
Ex: =TIME
15:33:34

YR

Will return the current year as an integer.
Ex: =YR

Sample Workspace

Below is a simple workspace that shows the use of different **KKEX** commands. Start **KKEX,** then in the Cell -> enter the value under CELL column and in the VALUE -> enter the value under the VALUE Column. After each entry **KKEX** will recompute the ENTIRE spreadsheet.

CELL	**VALUE**
A1	THIS IS A1
B1	=DATE
C1	=TIME
D1	14
Z1	THIS IS Z1
B2	=MTH
C2	=HOUR
D2	29
J2	THIS IS J2
W2	THIS IS W2
A3	=18+6
B3	=DAY
C3	=MIN
D3	89
B4	=YR
C4	=SEC
D4	=SUM(D1,D3)
E5	THIS IS E5
C6	=AVG(C2,C4)
D6	=D1+D2+D3
N6	THIS IS N6
10	THIS IS R10
15	THIS IS Z15

Installation

KKEX comes with two program files:

```
     kkgpm.c        KKEX program
     kkmsd.mapsrc   KKEX BMS Map
```

Upload these two programs to your KIKCS machine. The userid in both files is set to KICKS.

Before you load and/or compile the kk program/maps, you must link KICKS with GCC. At the CMS prompt, type the following command:

```
     link gcccms 591 591 rr
     acc 591 z
```

You need to only do this when you ipl cms.

Compile the BMS MAP (kkmsd.mapsrc) with the kikmg command.

```
     Ex:  kikmg kkmsd
```

Compile the KKEX program (kkpgm.c) with the kikgcccl command.

```
     Ex:  kikgcccl kkpgm
```

Next you will have to update two KICKS tables:

```
     kikppt1$ assemble.
     In this table you will make two entries, for the kkpgm and
     kkmsd map.
          KIKPPT TYPE=ENTRY,PROGRAM=KKPGM,PGMLAND-CMDLVL
          KIKPPT TYPE=ENTRY,PROGRAM=KKMSD,USAGE=MAP
     Next compile the table with the command:
          Ex:   kiktable ppt 1$


     kikpct1$ assemble
     In this table you will make only one entry, for kkpgm.
          KIKPCT TYPE=ENTRY,TRANSID=KKEX,PROGRAM=KKPGM
     Next compile the talbe with the command:
          Ex:   kiktable pct 1$
```

Start the KICKS system.

When KICKS is up, enter the transid of **KKEX**.
The KKEX program should load and be ready for use.