

"""

OLIVETTIZE.PY

INPUT A TEXT ASCII IMAGE AND GENERATE THE SIMULATED PRINTOUT USING CHARACTER IMAGES
BASED ON EARLIER PROGRAMS "MAKEIMAGE2.PY" AND "IBM1403.PY".

E.M.F. JANUARY 2021

1/5/2021 IMPLEMENTED IMAGE BLENDING FOR OVERSTRIKE OF SINGLE CHARACTERS
1/11/2021 ADDED NOTES ON PAGE SIZE; SAVE IMAGES AS 1200DPI
1/17/2021 REWRITTEN TO USE AN OBJECT CLASS "PRINTPAGE", ENABLE PRINTING TITLE AND
1/22/2021 REMOVED CODE TO PRINT AT BOTTOM OF PAGE
INTRODUCED PAGEROWS AND PAGECOLS, ABLE TO PRINT MULTIPLE PAGES
1/25/2021 CLEANED UP PAGE HANDLING AND CORRECTED THE PAGE FLUSH

"""

```
FROM PIL IMPORT IMAGE, IMAGEENHANCE, IMAGECHOPS
IMAGE.MAX-IMAGE-PIXELS = 110486672 # MY IMAGE SIZE EXCEEDS THE DOS WARNING
IMPORT SYS, MATH
```

"""

A PRINTPAGE IS AN IMAGE OBJECT THAT REPRESENTS AN 8.5X11" PAPER AT 1200DPI.
THE CURSOR IS STORED BY COLUMN AND ROW.
METHODS OF THE CLASS INCLUDE ADDING ("PRINTING") A CHARACTER IMAGE TO THE PAGE.

"""

CLASS PRINTPAGE:

```
# IBM1403 PAPER IS 14-7/8" WIDE AND 11" HIGH
# OLIVETTI TE-318 IS 8.5" WIDE AND 11" HIGH WITH A PRINTABLE AREA
PAGEROWS = 66
PAGECOLS = 85
PAGEWIDTH = PAGECOLS*120
PAGEHEIGHT = PAGEROWS*200 # THIS WAS 201
```

DEF --INIT--(SELF,NUMBER):

```
# PAGE NUMBER
SELF.NUMBER = NUMBER
# MAKE BLANK IMAGE PAGE (10CHARS/IN, 6 ROWS/IN)
MODE, SIZE, COLOR = 'L', (SELF.PAGEWIDTH, SELF.PAGEHEIGHT), 255
SELF.IMG = IMAGE.NEW(MODE, SIZE, COLOR)
```

START AT TOP LEFT CORNER OF PAGE

SELF.COLUMN = 0

SELF.ROW = 0

INITIAL OFFSET

IF NUMBER == 1: SELF.VOFFSET = 2

ELSE: SELF.VOFFSET = 0

SELF.HOFFSET = 3

DEF PRINTCHAR(SELF,CHAR):

```
# I HAVE 4 VERSIONS OF THE CHARACTER SET FOR SOME VARIATION, SO FILES ARE
# NN = 0-3, NN = ASCII VALUE
```

IF CHAR != ' ': # SPACE CHARACTER, JUST ADVANCE

```
CHARFILE = "CHARS/" + "[0:02N]".FORMAT(SELF.ROW % 3) + "[0:02N]".FORMAT
TRY:
```

```
WITH IMAGE.OPEN(CHARFILE) AS CHARIMG:
```

```
WIDTH, HEIGHT = CHARIMG.SIZE
```

```
IMGX = (SELF.COLUMN+SELF.HOFFSET)*WIDTH
```

```
IMGY = (SELF.ROW+SELF.VOFFSET)*HEIGHT
```

```
TEMPIMG = SELF.IMG.CROP((IMGX,IMGY,IMGX+WIDTH,IMGY+HEIGHT))
```

```
TEMPIMG = IMAGECHOPS.DARKER(TEMPIMG, CHARIMG)
```

```
SELF.IMG.PASTE(TEMPIMG,(IMGX,IMGY,IMGX+WIDTH,IMGY+HEIGHT))
```

EXCEPT OSERROR:

```
PRINT("ERROR OPENING CHARACTER FILE", CHARFILE, " LINE ",SELF.ROW
```

```
IF SELF.COLUMN <= 79: SELF.COLUMN += 1
```

```
"""
```

```
..MAIN PROGRAM
```

```
"""
```

```
IF <<NAME<< == '<<MAIN<<':
```

```
THE TEXT FILE FOR AN ASCII IMAGE IS TYPICALLY 132 COLUMNS WIDE AND 66 ROWS LONG  
THE PAPER IS 14 7/8" WIDE AND 11" LONG, SO THE INITIAL INDENT SHOULD BE 8 CHARACTERS
```

```
# I'M EXPECTING THE TEXT FILE AS THE FIRST ARGUMENT
```

```
# OPEN THE FILE AS A BINARY TO CAPTURE THE CR'S
```

```
IF LEN(SYS.ARGV) == 2:
```

```
    FILENAME = SYS.ARGV[1]
```

```
    TRY:
```

```
        WITH OPEN(FILENAME, MODE='rb') AS TEXTFILE:
```

```
            # INITIALIZE THE PAGE
```

```
            NUMPAGE = 1
```

```
            PAGE = PRINTPAGE(NUMPAGE)
```

```
            OUTPUT = []
```

```
            # PROCESS EACH LINE OF THE TEXT FILE
```

```
            # REWRITE TO USE THE FORTRAN CARRIAGE CONTROL CHARACTERS
```

```
            FOR LINE IN TEXTFILE:
```

```
                # PROCESS EACH CHARACTER IN LINE
```

```
                FOR BYTE IN LINE:
```

```
                    IF BYTE == 10: # NEW LINE
```

```
                        PAGE.ROW += 1
```

```
                        PAGE.COLUMN = 0
```

```
                        IF PAGE.ROW == PAGE.PAGEROWS - PAGE.VOFFSET:
```

```
                            OUTPUT.APPEND(PAGE)
```

```
                            NUMPAGE += 1 # ADD A PAGE
```

```
                            PAGE = PRINTPAGE(NUMPAGE)
```

```
                    ELIF BYTE == 13: # CARRIAGE RETURN - OVERSTRIKE
```

```
                        PAGE.COLUMN = 0
```

```
                    ELIF BYTE == 12: # FORM FEED
```

```
                        OUTPUT.APPEND(PAGE)
```

```
                        NUMPAGE += 1
```

```
                        PAGE = PRINTPAGE(NUMPAGE)
```

```
                    ELSE:
```

```
                        # DRAW EACH CHARACTER ON THE PAGE
```

```
                        CHAR = INT(BYTE)
```

```
                        IF CHAR > 95: CHAR -= 32 # OLIVETTI ONLY PRINTS 32-95
```

```
                        PAGE.PRINTCHAR(CHAR)
```

```
            # FLUSH THE LAST PAGE TO THE OUTPUT
```

```
            OUTPUT.APPEND(PAGE)
```

```
            PDF = []
```

```
            FOR PAGE IN OUTPUT:
```

```
                PRINT("PRINTING PAGE ", PAGE.NUMBER, " OF ", NUMPAGE)
```

```
                PAGE.IMG = PAGE.IMG.RESIZE((PAGE.PAGEWIDTH//2, PAGE.PAGEHEIGHT//2))
```

```
                PDF.APPEND(PAGE.IMG)
```

```
                PAGE.IMG.SAVE(FILENAME.SPLIT('.')[0] + '-0-' + STR(PAGE.NUMBER) + '.img')
```

```
            PRINT("SAVING PDF...")
```

```
            PDF[0].SAVE(FILENAME.SPLIT('.')[0] + '.PDF', "PDF", RESOLUTION=600)
```

```
            TEXTFILE.CLOSE()
```

```
        EXCEPT OSERROR:
```

```
            PRINT("ERROR OPENING", FILENAME)
```

```
    ELSE:
```

```
        PRINT("\NUSAGE: OLIVETTIZE.PY FILENAME \N")
```