FILE INPUT/OUTPUT*

*** Some of the contents are adopted from MATLAB® for Engineers, Holly Moore, 3rd Ed., Pearson Inc., 2012.**

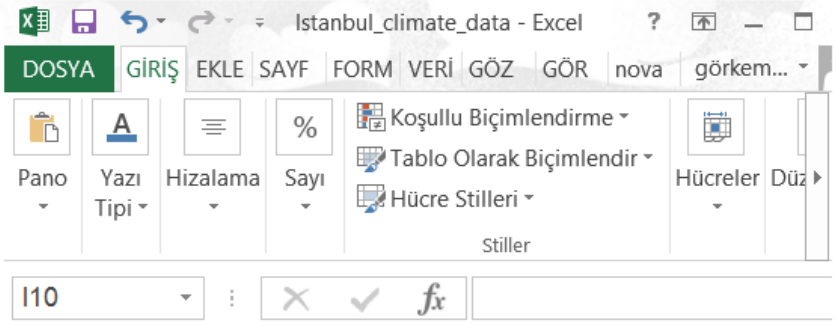
File Input/Output

- File:
  - Area in permanent storage (disk drive)
  - Stores information
  - Managed by the operating system
  - Can be copied or moved
  - Can be accessed by programs
- File Input/Output (I/O)
  - Data exchange between programs and computers
  - Data exchange between the physical world and computers
  - Saving your work so you can continue with it later
- MATLAB can handle
  - Mat-files and M-files <u>AND</u> text, binary, and Excel files

Excel files

- Microsoft Excel® is a widely used data-analysis tool
- Many other programs support reading and writing Excel files
- MATLAB does too with two built-in functions
  - **`xlsread`**
  - **`xlswrite`**

Reading Excel files



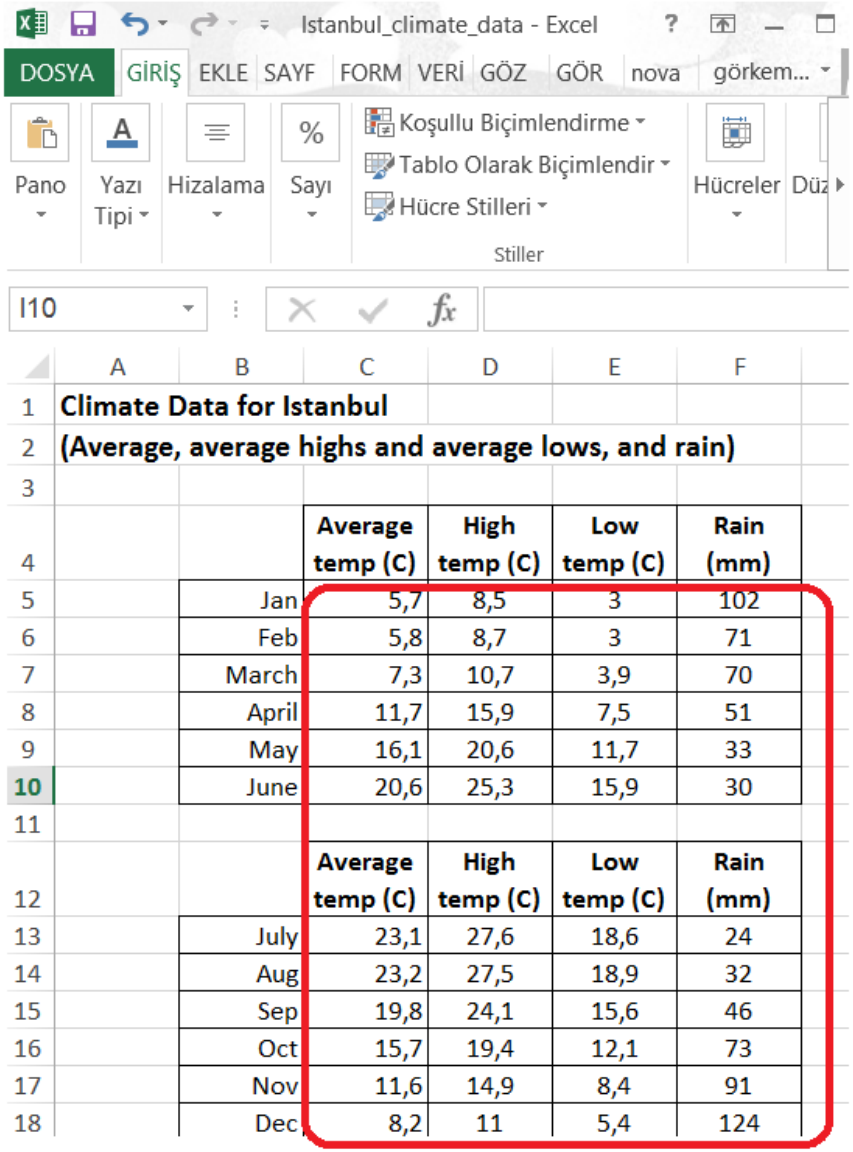
	A	B	C	D	E	F
1	**Climate Data for Istanbul**					
2	**(Average, average highs and average lows, and rain)**					
3						
4			**Average temp (C)**	**High temp (C)**	**Low temp (C)**	**Rain (mm)**
5		Jan	5,7	8,5	3	102
6		Feb	5,8	8,7	3	71
7		March	7,3	10,7	3,9	70
8		April	11,7	15,9	7,5	51
9		May	16,1	20,6	11,7	33
10		June	20,6	25,3	15,9	30
11						
12			**Average temp (C)**	**High temp (C)**	**Low temp (C)**	**Rain (mm)**
13		July	23,1	27,6	18,6	24
14		Aug	23,2	27,5	18,9	32
15		Sep	19,8	24,1	15,6	46
16		Oct	15,7	19,4	12,1	73
17		Nov	11,6	14,9	8,4	91
18		Dec	8,2	11	5,4	124

```
[num,txt,raw] = xlsread('Istanbul_climate_data.xlsx');
```

Numerical

rium =

5.7000	8.5000	3.0000	102.0000
5.8000	8.7000	3.0000	71.0000
7.3000	10.7000	3.9000	70.0000
11.7000	15.9000	7.5000	51.0000
16.1000	20.6000	11.7000	33.0000
20.6000	25.3000	15.9000	30.0000
NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN
23.1000	27.6000	18.6000	24.0000
23.2000	27.5000	18.9000	32.0000
19.8000	24.1000	15.6000	46.0000
15.7000	19.4000	12.1000	73.0000
11.6000	14.9000	8.4000	91.0000
8.2000	11.0000	5.4000	124.0000



Climate Data for Istanbul

(Average, average highs and average lows, and rain)

	Average temp (C)	High temp (C)	Low temp (C)	Rain (mm)
Jan	5,7	8,5	3	102
Feb	5,8	8,7	3	71
March	7,3	10,7	3,9	70
April	11,7	15,9	7,5	51
May	16,1	20,6	11,7	33
June	20,6	25,3	15,9	30

	Average temp (C)	High temp (C)	Low temp (C)	Rain (mm)
July	23,1	27,6	18,6	24
Aug	23,2	27,5	18,9	32
Sep	19,8	24,1	15,6	46
Oct	15,7	19,4	12,1	73
Nov	11,6	14,9	8,4	91
Dec	8,2	11	5,4	124

Text

```
txt =

    [1x25 char]    ''          ''                      ''                  ''                  ''
    [1x51 char]    ''          ''                      ''                  ''                  ''
    ''             ''          ''                      ''                  ''                  ''
    ''             ''          'Average temp (C)'      'High temp (C)'     'Low temp (C)'      'Rain (mm)'
    ''             'Jan'       ''                      ''                  ''                  ''
    ''             'Feb'       ''                      ''                  ''                  ''
    ''             'March'     ''                      ''                  ''                  ''
    ''             'April'     ''                      ''                  ''                  ''
    ''             'May'       ''                      ''                  ''                  ''
    ''             'June'      ''                      ''                  ''                  ''
    ''             ''          ''                      ''                  ''                  ''
    ''             ''          'Average temp (C)'      'High temp (C)'     'Low temp (C)'      'Rain (mm)'
    ''             'July'      ''                      ''                  ''                  ''
    ''             'Aug'       ''                      ''                  ''                  ''
    ''             'Sep'       ''                      ''                  ''                  ''
    ''             'Oct'       ''                      ''                  ''                  ''
    ''             'Nov'       ''                      ''                  ''                  ''
    ''             'Dec'       ''                      ''                  ''                  ''
```

All data: cell array

```
raw =

    [1x25 char]    [  NaN]    [             NaN]    [           NaN]    [          NaN]    [       NaN]
    [1x51 char]    [  NaN]    [             NaN]    [           NaN]    [          NaN]    [       NaN]
    [      NaN]    [  NaN]    [             NaN]    [           NaN]    [          NaN]    [       NaN]
    [      NaN]    [  NaN]    'Average temp (C)'    'High temp (C)'    'Low temp (C)'    'Rain (mm)'
    [      NaN]    'Jan'      [          5.7000]    [        8.5000]    [           3]    [       102]
    [      NaN]    'Feb'      [          5.8000]    [        8.7000]    [           3]    [        71]
    [      NaN]    'March'    [          7.3000]    [       10.7000]    [       3.9000]    [        70]
    [      NaN]    'April'    [         11.7000]    [       15.9000]    [       7.5000]    [        51]
    [      NaN]    'May'      [         16.1000]    [       20.6000]    [      11.7000]    [        33]
    [      NaN]    'June'     [         20.6000]    [       25.3000]    [      15.9000]    [        30]
    [      NaN]    [  NaN]    [             NaN]    [           NaN]    [          NaN]    [       NaN]
    [      NaN]    [  NaN]    'Average temp (C)'    'High temp (C)'    'Low temp (C)'    'Rain (mm)'
    [      NaN]    'July'     [         23.1000]    [       27.6000]    [      18.6000]    [        24]
    [      NaN]    'Aug'      [         23.2000]    [       27.5000]    [      18.9000]    [        32]
    [      NaN]    'Sep'      [         19.8000]    [       24.1000]    [      15.6000]    [        46]
    [      NaN]    'Oct'      [         15.7000]    [       19.4000]    [      12.1000]    [        73]
    [      NaN]    'Nov'      [         11.6000]    [       14.9000]    [       8.4000]    [        91]
    [      NaN]    'Dec'      [          8.2000]    [            11]    [       5.4000]    [       124]
```

Text files

- Text files contain characters
- They use an encoding scheme:
  - ASCII or
  - Any one of many other schemes
  - MATLAB takes care of encoding and decoding
- Before using a text file, we need to open it
- Once done with the file, we need to close it

Opening text files

- Opening: `fid = fopen(filename, permission)`
- Closing: `fclose(fid)`
- `fid`: Unique file identifier for accessing file
- Permission: what we want to do with the file—
  - read, write, overwrite, append, etc.

2ND ARGUMENT	PERMISSION
`'rt'`	open text file for reading
`'wt'`	open text file for writing; discard existing contents
`'at'`	open or create text file for writing; append data to end of file
`'r+t'`	open (do not create) text file for reading and writing
`'w+t'`	open or create text file for reading and writing; discard existing contents
`'a+t'`	open or create text file for reading and writing; append data to end of file

Reading text files

- One line at a time
- `type` prints a text file in the command window
- Let's re-implement it:

```
function view_text_file(filename)
fid = fopen(filename,'rt');
if fid < 0
    error('error opening file %s\n\n', filename);
end

% Read file as a set of strings, one string per line:
oneline = fgets(fid);
while ischar(oneline)
    fprintf('%s',oneline) % display one line
    oneline = fgets(fid);
end
fprintf('\n');
fclose(fid);
```

Reading text files

- Reading lines into string variables is easy
- Parsing these strings to get numerical data is much harder
- Not covered
- Binary files are more suited for numerical data

Binary files

- Binary file = “not a text file”
- Many different ways to represent numbers
- All we need to know are their types.
- Binary files need to be
  - Opened with **`fopen`**
  - Closed with **`fclose`**

2ND ARGUMENT	PERMISSION
`'r'`	open binary file for reading
`'w'`	open binary file for writing; discard existing contents
`'a'`	open or create binary file for writing; append data to end of file
`'r+'`	open (do not create) binary file for reading and writing
`'w+'`	open or create binary file for reading and writing; discard existing contents
`'a+'`	open or create binary file for reading and writing; append data to end of file

Writing binary files

- Data type is important
- Example: write a double array into a binary file

```
function write_array_bin(A,filename)
fid = fopen(filename,'w+');
if fid < 0
    error('error opening file %s\n', filename);
end

fwrite(fid,A,'double');

fclose(fid);
```

Reading binary files

- Example: read a double array from a binary file

```
function A = read_bin_file(filename,data_type)
fid = fopen(filename,'r');
if fid < 0
    error('error opening file %s\n',filename);
end

A = fread(fid,inf,data_type);

fclose(fid);
```

Example 1

- Write a function called **`letter_counter`** that takes the name of a text file as input and returns the number of letters (i.e., any of the characters, a-to-z and A-to-Z) that the file contains. HINT: You can use the built-in function **`isletter`**. If there is a problem opening the file, the function returns -1. **WARNING**: if you use the ‘w’ flag with **`fopen`**, as opposed to ‘r’, you will overwrite the file.

Example 1

```
function n = letter_counter(fname)
    fid = fopen(fname,'r');
        if fid < 0
            n = -1;
        else
            x = fread(fid,inf,'char'); % read entire file
            x = x(isletter(char(x))); % pick the letters
            n = length(x); % count them
            fclose(fid);
        end
end
```