

MATLAB Unit 2-Lecture 4

BTech (CSBS) -Semester VII

22 July 2022, 09:35AM



- Managing the workspace
- Keeping track of your work session,
- Entering multiple statements per line

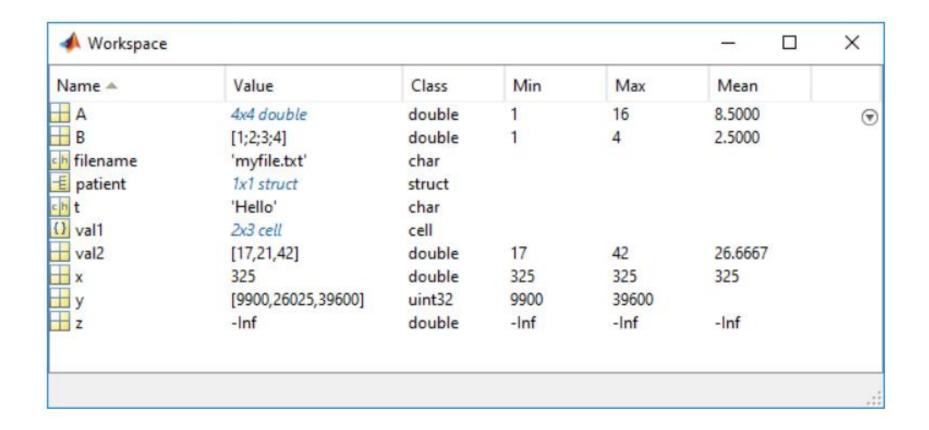


Workspace Browser

- The Workspace browser enables you to view and interactively manage the contents of the workspace in MATLAB®.
- For each variable or object in the workspace, the Workspace browser also can display statistics, when relevant, such as the minimum, maximum, and mean.
- You can edit the contents of scalar (1-by-1) variables directly in the Workspace browser. Right-click the variable and select Edit Value.
- To edit other variables, double-click the variable name in the Workspace browser to open it in the Variables editor.



Workspace Browser





Open the Workspace Browser

To open the Workspace browser if it is not currently visible, do one of the following:

- MATLAB Toolstrip: On the **Home** tab, in the **Environment** section, click **Layout**. Then, in the **Show** section, select **Workspace**.
- MATLAB command prompt: Enter workspace.

You also can minimize the Workspace browser by collapsing the panel in which it resides. eg. if the Workspace browser is in the left side panel, click the button at the bottom left corner of the panel to collapse the panel. To restore the panel, click the button. If the Workspace browser is in the left or right side panel and the panel contains multiple tools, you also can minimize it by clicking the button to the left of the Workspace browser title bar.



Workspace Variables

Functions

load	Load variables from file into workspace
save	Save workspace variables to file
matfile	Access and change variables in MAT-file without loading file into memory
disp	Display value of variable
formattedDisplayText	Capture display output as string
who	List variables in workspace
whos	List variables in workspace, with sizes and types
clear	Remove items from workspace, freeing up system memory
clearvars	Clear variables from memory
openvar	Open workspace variable in Variables editor or other graphical editing tool
Workspace Browser	Open Workspace browser to manage workspace



Save Workspace Variables

There are several ways to save workspace variables interactively:

- To save all workspace variables to a MAT-file, on the **Home** tab, in the **Variable** section, click **Save Workspace**.
- To save a subset of your workspace variables to a MAT-file, select the variables in the Workspace browser, right-click, and then select **Save As**. You also can drag the selected variables from the Workspace browser to the Current Folder browser.
- To save variables to a MATLAB script, click the **Save Workspace** button or select the **Save As** option, and in the **Save As** window, set the **Save as type** option to **MATLAB Script.** Variables that cannot be saved to a script are saved to a MAT-file with the same name as that of the script.

Save Workspace Variables

You also can save workspace variables programmatically using the **save** function.

save('june10')

To save only variables A and B to the file june 10.mat, use the command

save('june10', 'A', 'B')



Load Workspace Variables

- To load saved variables from a MAT-file into your workspace, double-click the MAT-file in the Current Folder browser.
- To load a subset of variables from a MAT-file on the **Home** tab, in the **Variable** section, click **Import Data**. Select the MAT-file you want to load and click **Open**. You also can drag the desired variables from the Current Folder browser Details panel of the selected MAT-file to the Workspace browser. The Details panel is not available in MATLAB Online.
- To load variables saved to a MATLAB script into the workspace, simply run the script.

Load Workspace Variables

You also can load saved variables programmatically, use the load function.

load('durer')

To load variables X and map from the file durer.mat

load('durer', 'X', 'map')



Keep a track of work

Write to a Diary File

To keep an activity log of your MATLAB® session, use the **diary** function. diary creates a verbatim copy of your MATLAB session in a disk file (excluding graphics).



entering multiple statements per line

- Enter Multiple Lines Without Running Them
- To enter multiple lines before running any of them, use **Shift+Enter** or **Shift+Return** after typing a line. This is useful, for example, when entering a set of statements containing keywords, such as if ... end. The cursor moves down to the next line, which does not show a prompt, where you can type the next line. Continue for more lines. Then press **Enter** or **Return** to run all of the lines.
- This allows you to edit any of the lines you entered before you pressing Enter or Return.
- Entering Multiple Functions in a Line
- To enter multiple functions on a single line, separate the functions with a **comma** (,) or **semicolon** (;). Using the semicolon instead of the comma will suppress the output for the command preceding it. For example, put three functions on one line to build a table of logarithms by typing

```
format short; x = (1:10)'; logs = [x log10(x)]
```

and then press Enter or Return. The functions run in left-to-right order.



entering multiple statements per line

- Entering Long Statements
- For items in single quotation marks, such as strings, you must complete the string in the line on which it was started. For example, completing a string as shown here

```
headers = ['Author Last Name, Author First Name, ' ... 'Author Middle Initial']
results in
```

headers = Author Last Name, Author First Name, Author Middle Initial