GETTING STARTED QUESTIONS: Getting started

FOCUS QUESTION: How do I start using MATLAB?

Contents

EXAMPLE 1: Read data from a file into the MATLAB workspace

```
load count.dat;
```
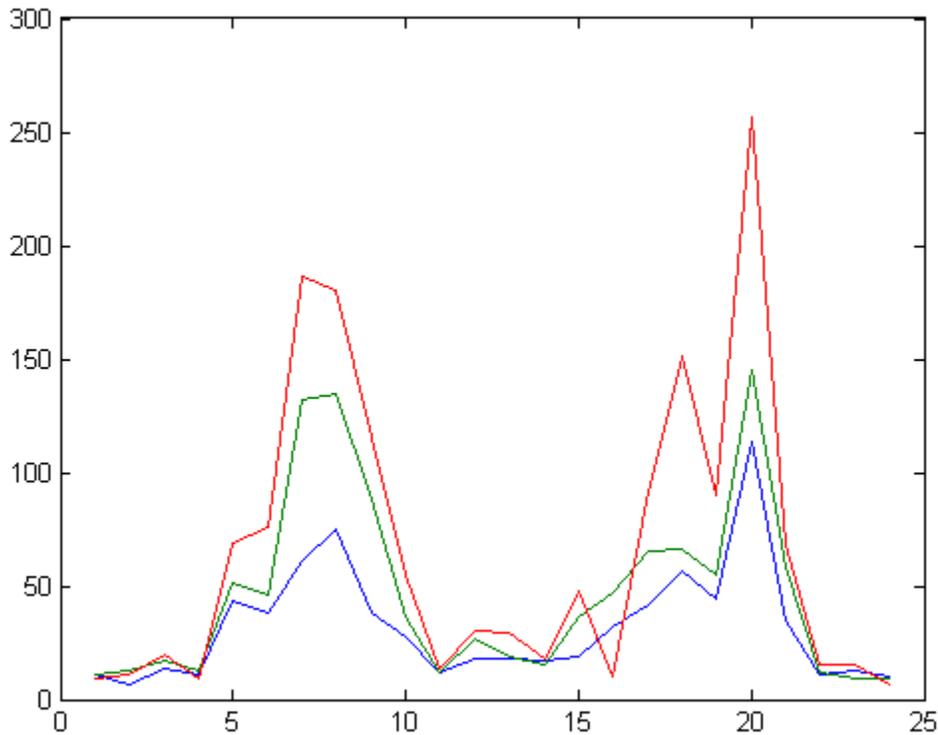
Questions	Answers
How do I read data into MATLAB?	The simplest way is to use the `load` command.
What happens when MATLAB executes `load`?	The `load` command reads data from a file on disk. If the file is a `.mat` file, the `load` command reads previously saved MATLAB data into the workspace and recreates the variables that were used to name or identify this data. The `load` command also works for text files such as `count.dat`. In this case the values are assigned to a single array named by the file name (without the extension).
Can I use `load` to read any type of data?	The `load` command works for `.mat` files and for plain text files that are in certain formats. We will learn other commands and functions for reading other types of data.
Where did the `count.dat` file come from?	The `count.dat` file is used for several MATLAB demos and comes with the MATLAB distribution. It should always be available where MATLAB is.
What is `count`?	`count` is a MATLAB variable.
What is a variable and what good is it?	A variable is a way of holding data in a program. Variables have names and values. The value of the variable is the data that the variable holds. You can refer to the data by the variable's name. You can change the value of a variable by assigning it a new value (hence the name *variable*).
What data does the `count` variable identify?	The `count` variable names a table or array that holds traffic data for 3 intersections measured hourly over a 24-hour period.
What is the Workspace Browser?	The Workspace Browser shows all of the variables currently defined in MATLAB's "workspace".
Why does the Workspace Browser show `count` as 24x3 double?	The `count` variable holds an array of real values arranged in 24 rows and 3 columns.

Questions	Answers
Why don't I see the actual values of `count` **in the Workspace Browser?**	The table of data held by `count` is too large to display in the space available in the Workspace Browser.
How do I see the values of `count`**?**	Double click on the name `count` in the Workspace Browser to bring up the Variable Editor.
If I make a change to `count` **in the Variable Editor, will** `count` **be changed permanently?**	Yes, the changes are permanent as soon as you make them. However, these changes are not made to the copy of `count.dat` on disk unless you explicitly save them.
What does the semicolon (`;` **) at the end of the** `load` **command do?**	The semicolon ends a command and suppresses the output if there is any.
Do I need a semicolon to end every command?	No, MATLAB treats the end of the line as the end of the command.
What if my command takes up more than one line?	Use three periods in a row (`...`) at the end of the line to continue the command on the next line.

EXAMPLE 2: Plot the data just read

```
figure

plot(count)
```

Questions	Answers
What does `plot(count)` do?	This MATLAB command plots each column of `count` as an individual line graph. MATLAB plots the values against the numbers 1, 2, 3, 4, ..., 24, because `count` has 24 rows. Since `count` has three columns, you see three separate line graphs. The three graphs appear on the same figure.
What does the horizontal axis on this graph represent?	The horizontal axis represents the positions or row numbers within the data, starting with row 1.
What happens if I omit the `figure` command?	If a current Figure Window already exists, the plot overwrites the plot displayed in that window. Otherwise, MATLAB automatically creates a new Figure Window. Use `figure` each time you plot in order to see all of the figures that you make.
Why don't these commands end with a semicolon?	Neither the `figure` nor the `plot` command produce any output, so it isn't necessary to end them with a semicolon.

EXAMPLE 3: Plot the data in an informative way

Type the following in the lesson 1 script, save, and run:

```
figure

plot(count)
```

```
    title('Traffic at 3 San Antonio intersections')

    xlabel('Hour of day (24 hour format)')

    ylabel('Number of vehicles')

    legend('Durango','DeZavala', 'LP410')
```
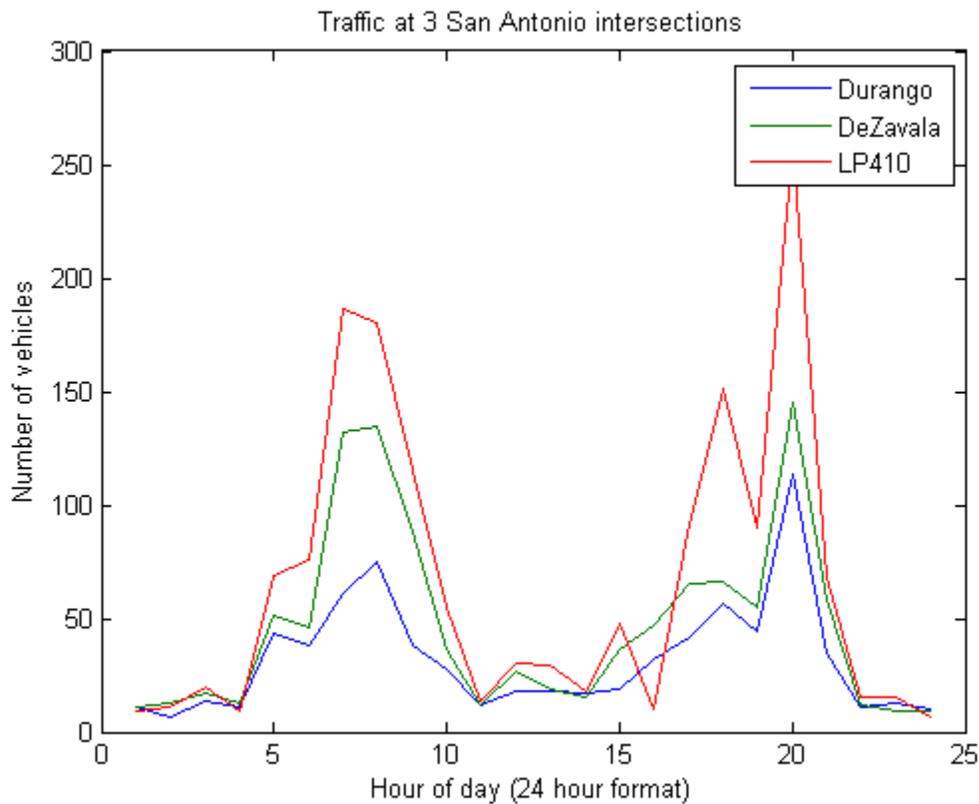
You should see a Figure Window with a labeled plot of the traffic:

Questions	Answers
What is `xlabel`?	The `xlabel` identifier names a MATLAB function that sets the x-axis label on the current axis. Similarly, the `ylabel` function sets the y-axis label on the current axis, and the `title` function sets a title over the current axis.
Why call a function such as `xlabel` rather than editing the x-axis directly using the plot tools?	Calling `xlabel`, `ylabel`, and `title` documents the purpose of the graph in your script as well as labeling the graph. If you need to make a change on the graph, you simply change the script and rerun rather than recreating a new plot manually.
What is a legend?	A legend is an annotation on a plot identifying the type of data represented by objects in the plot.

Questions	Answers
What does the word `legend` represent in MATLAB?	The `legend` identifier names a MATLAB function that provides annotations for the objects on a graph. The function arguments (the items in parentheses) specify how to identify the plot objects in the annotation.
Why is `'DeZavala'` enclosed in quotes?	MATLAB uses single quotes to distinguish strings or labels from variable and function names. Without the quotes, `DeZavala` would be a variable and the value it holds would be used.
When should I use a legend?	If your axes have more than one graph, you should always use a legend. By calling the `legend` function rather than waiting to edit the plot later, you will provide documentation for your graph in your script.

EXAMPLE 4: Reorganize your `lesson1Script` file into cells

Questions	Answers
What is cell mode?	Cell mode is a way of developing and executing MATLAB scripts in the Editor. The script is organized into subsections or cells, which you can execute individually.
What is a cell?	A cell is a group of statements in a MATLAB script that starts with "`%%` " and ends with the start of the next cell. You can execute the statements within a cell as a unit, separately from the rest of the script.
Why is cell mode useful?	Cell mode allows you to develop scripts in a step-by-step manner and test each step for correctness. This strategy makes it much easier to develop code that works.
How do I put in cell dividers?	Each time you insert a line that starts with `%%` followed by at least one blank, you start a cell. You can also use `Cell->Insert Cell Break` from the MATLAB menu bar or you can press the insert cell icon on the Editor toolbar.
How do I evaluate or execute a cell?	You can right click anywhere in the cell and choose `Cell->Evaluate Current Cell` from the MATLAB menubar. You can also use the Editor icons or to execute the current cell.
What determines the current cell?	The current cell is where your cursor is. (It should be highlighted in pale yellow.)
How do I start a new cell?	Create a line starting with two percent signs followed by a space. After placing the cursor where you want to insert the cell, you can either type these characters on the keyboard or hit the Insert cell break icon .
What if I forget the space?	The lines that follow will be part of the previous cell.
What does the single percent sign designate?	The `%` marks the start of a comment.
Does MATLAB execute comments?	No, comments are for the benefit of the user and are ignored during execution.

EXAMPLE 5: Publish your final script.

Questions	Answers
What is the Command Window?	The Command Window is a place where you can directly enter and execute MATLAB commands. The Command Window also displays errors generated when you execute a script.
What is a script?	A script is a file containing MATLAB commands. Scripts should have the `.m` file extension.
What is an m-file?	An m-file is a special type of MATLAB file that has a `.m` file extension. You can store a list of MATLAB commands (a script) as an m-file. Later you will also create your own functions and save them as m-files.
What is the difference between an m-file and a published script?	An m-file contains commands that MATLAB can execute. A published script is a document that shows the results of running the script in a readable format. By default, the published version appears as a web page, but other forms are possible.
Why bother saving commands in an m-file?	These saved commands can be reused and modified, saving time and effort. Scripts allow you to automate data analysis when you have a large number of similar data sets. With scripts you can develop complicated analysis in a step-by-step manner and the script documents your steps.
Can I add commands from the Command History window into an existing m-file?	You can either drag commands from the Command History window into the m-file, or you can use cut and paste.
Can I create an m-file from the Command History window?	You can highlight commands in the Command History window, right click, and choose `Create m-file`. As with other applications, use `Shift-click` to add items to the selection and `Ctrl-click` to remove items from the selection.

These questions were written by Kay A. Robbins of the University of Texas at San Antonio and last updated by Dawn Roberson on 3-Jan-2014. Please contact Kay.Robbins@utsa.edu with comments or suggestions.