**INTRODUCCIÓN A MATLAB**

MATLAB (MAtrix LABoratory) es un programa cuyo propósito es la optimización de los cálculos científicos realizados en ingeniería, inicialmente concebido para el trabajo con matrices y vectores, pero a lo largo del tiempo ha evolucionado en un sistema de cómputo flexible capaz de resolver esencialmente cualquier problema de tipo técnico.

MATLAB posee un lenguaje de programación además de una extensa librería de funciones predeterminadas que le permiten realizar las tareas de cálculo más complicadas de una forma sencilla.

MATLAB es un programa gigante, sólo su versión básica posee cerca de 100 funciones, y sus toolboxes (cajas de herramientas) extienden su capacidad a muchas funciones especializadas.

MATLAB posee toolboxes casi para cada rama de la ingeniería.

Ejemplos de algunos toolbox:

|  |  |  |
| --- | --- | --- |
| MATLAB environment | Fuzzy Logic | Bioinformatics |
| Simulink | Global Optimization | Curve Fitting |
| Control System | Image Acquisition | Data Acquisition |
| Financial | Instrument Control | Database |
| Image Processing | Mapping | DSP System Toolbox |
| Optimization | MATLAB Coder | Econometrics |
| Signal Processing | MATLAB Compiler | Embedded Coder |
| Statistics and Machine Learning | Neural Network | Filter Design HDL Coder |
| System Identification | Parallel Computing | Financial Instruments |
| Wavelet | Partial Differential Equation | Fixed-Point Designer Toolbox |
| SimMechanics | Simulink Control Design | Simulink 3D Animation |
| Simscape | Stateflow | Spreadsheet Link EX |
| Simulink Coder | Symbolic Math | Simulink Real-Time |

Cada uno de estos toolbox tiene su propio conjunto de instrucciones.

Ventajas de MATLAB:

* Es fácil de usar, su lenguaje de programación es interpretado, lo que facilita la ejecución de sus códigos.
* Es multiplataforma, viene en versiones para Windows, OS Mac, Unix y Linux, los programas escritos en una plataforma funcionarán en cualquiera de ellas.
* Sus funciones predefinidas le permiten ejecutar comandos en poco tiempo.

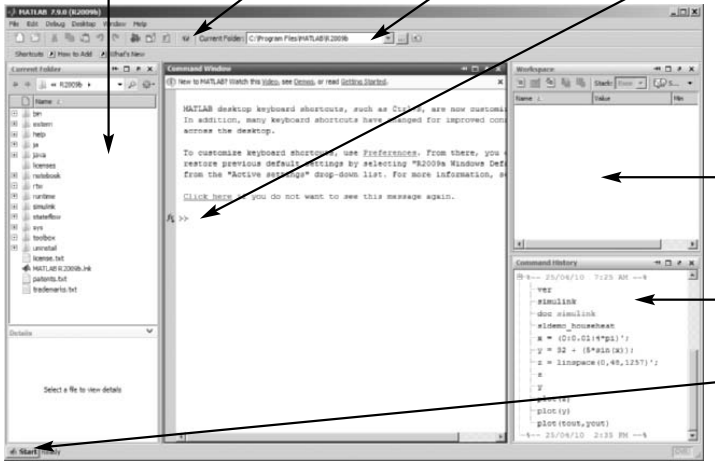
Desventajas de MATLAB:

* Por ser un lenguaje de programación interpretado sus códigos demoran más en ser ejecutados.
* El costo, la versión básica de MATLAB cuesta USD $85.00, más USD $45.00 cada toolbox, instalar una versión completa costaría cerca de USD $2,000.00 solamente en un equipo.

Por ello existen versiones estudiantiles que no tienen todas las funcionalidades de la versión profesional.

Contenido Ventana de comando o

Carpeta actual ayuda en web Carpeta actual Command Window



Workspace

o espacio de trabajo

Botón de inicio

Command history

o historial de

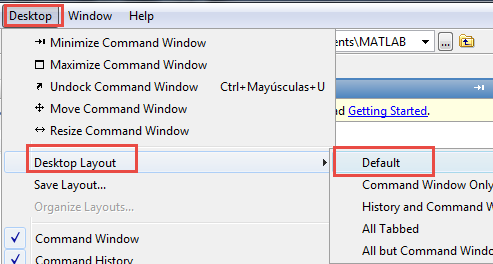
comandos

|  |  |
| --- | --- |
| Command window | Es una ventana donde el usuario puede digitar los comandos y ver los resultados inmediatos. |
| Command history window | Es un panel donde se muestran los comandos usados recientemente. |
| Botón de inicio (Start button) | Permite el acceso inicial a MATLAB, a sus toolboxes y de más recursos. |
| Contenido carpeta actual | Muestra los archivos que se encuentran en la carpeta a la cual apunta Matlab. |
| Workspace | Muestra las variables que están en uso en los programas que ha ejecutado Matlab en la sesión actual. |
| Carpeta Actual | Es la carpeta a la cual está apuntando Matlab para abrir o guardar archivos .M, el usuario puede escoger que carpeta utilizar, ya sea en una unidad externa o disco local. |

Por ahora veremos cómo introducir comandos en la **Command window** (ventana de comando).

Se caracteriza por los “ >> “, lo que significa que está listo para recibir comandos, a estos símbolos se les conoce también como **Prompt.**

¿Y si no aparecen los paneles de los que estamos hablando?



**Para detener procesos en Matlab:**

Existen ocasiones en las cuales el programa se sale de control, escribimos mal un código, lo que produce que Matlab tarde mucho en responder o produzca una serie interminable de números, P.E:

Al digitar en la ventana de comando **rand (1e7, 1),** causará que Matlab empiece a generar números aleatorios, para detener estas operaciones o desbloquear Matlab se puede utilizar el comando Ctrl C, lo que causa que el programa aborte cualquier proceso o programa en ejecución.

**Mensajes de estado de Matlab:**

|  |  |
| --- | --- |
| * Cuando el programa está listo para empezar, en la parte inferior izquierda, justo al lado del botón **START**, aparecerá el texto “**Ready**”. |  |
| * Cuando el programa está ejecutando códigos, o programas aparecerá la palabra “**Busy**”, que significa ocupado. |  |
| * Cuando al ejecutar un programa Matlab está esperando entradas o alguna interacción del usuario, dirá **“Waiting for input”,** que significa esperando una entrada. |  |

**Para cerrar Matlab por comandos:**

Se puede digitar el comando “**quit**”.

**Para limpiar la ventana de comandos, sin borrar los datos de memoria:**

Se puede digitar el comando “**clc**”.

**Para borrar todas las variables con sus datos:**

Se puede digitar el comando “**clear all**”.

**Primeros comandos en Matlab**

**Crear variables:**

Digitar en la ventana de comando la siguiente operación y luego la tecla enter:

|  |  |
| --- | --- |
|  |  |
| Asignar valor a x | Consultar valor de x |

Matlab crea la variable x de forma inmediata y al realizar la operación le asigna el valor de 3.

No olvidemos que las variables son simplemente una ubicación en la memoria del computador con un nombre y que guardan un valor, lo cual es diferente a la definición de variable desde el punto de vista matemático.

Al digitar de nuevo **x**, y dar enter, Matlab consultará el valor en la memoria.

Un error muy común en Matlab es utilizar o llamar una variable que no ha sido asignada, P.E:

Al digitar la variable que está vacía, no asignada o que no tiene ningún dato almacenado, aparecerá el error que se ve en el cuadro de abajo.

|  |  |
| --- | --- |
|  | Indefinida función o variable ‘ y ’. |

**Llamar valores o líneas ingresadas anteriormente en Matlab:**

En la ventana de comandose pueden pulsar las teclas de las flechas arriba y abajo para tener acceso a líneas que fueron previamente digitadas, presionar enter al seleccionar la línea deseada.



|  |
| --- |
| **Reglas para la designación de variables en Matlab**   * Los nombres de las variables no deben empezar por números P.E 1area   Los nombres pueden contener números o tenerlos al final del nombre P.E  area1triangulo, area1.   * Los nombres de las variables no deben contener caracteres especiales como tildes, signos de número (#), signos de pesos ($) o espacios finales o intermedios, si desea separar digite un guion inferior P.E area\_triangulo * Se recomiendan que los nombres de las variables sean en minúsculas. * Las variables no pueden tener nombres iguales a las funciones o palabras reservadas de Matlab. P.E: coseno -> sería una variable válida.   cos -> no lo sería porque es el nombre de una función de Matlab. |

**Actividad 1:**

1. Vamos a calcular la cantidad de kilómetros en un año luz.

Ingresamos las siguientes variables a la ventana de comando de Matlab:





Luego vamos a realizar la operación, vamos a multiplicar estas dos variables y la vamos a almacenar en la variable **year\_luzkm:**



Antes de terminar la **Actividad 1**, verifiquemos esto:

**Comentarios en Matlab**

Se pueden escribir notas aclaratorias o datos adicionales mediante comentarios, ellos se digitan precedidos del signo ‘ **%** ‘ , dichos comentarios no serán tenidos en cuenta por Matlab y aparecen en color verde.

**El comando Format**

Sirve para configurar como se muestran los resultados en la ventana de comando, observemos algunas configuraciones:

|  |  |
| --- | --- |
| **Format long** Es un comando que se utiliza para mostrar precisión hasta de 15 dígitos después de la coma. P.E | **Format short** Muestra 5 dígitos después de la coma. P.E |
| **Format compact** Sirve para ahorrar espacio en el momento de mostrar resultados en la ventana de comando. | **Format loose** agrega unas líneas extras al mostrar los resultados y es la forma predeterminada. |

Es importante tener en cuenta que format short y long no modifican como Matlab hace los cálculos, esto significa, que el programa siempre calculará con máximo 14 dígitos después de la coma, pero con format short sólo mostrará 4, en realidad el numero calculado es de mayor precisión, el usuario simplemente muestra como verlo.

1. Ahora, para terminar la **Actividad 1**, ¿cuál tipo de formato, vistos en el cuadro anterior, deberíamos aplicar para ver la cantidad de km de un año luz, suponiendo que queremos ver más dígitos en el resultado?

**Actividad 2:** Los números extensos en Matlab se pueden ingresar utilizando la notación científica, P.E: vamos a calcular el tiempo que tarda la luz de sol en llegar a la tierra, en segundos y minutos.

Vamos a ingresar la distancia entre la tierra y el sol en km, que es 150 millones.



Luego se divide utilizando la variable de la velocidad de la luz para obtener el resultado en segundos:



Terminar de calcular el tiempo en minutos.

**Actividad 3**: La distancia entre la tierra y la luna es de 384400 km, calcular cuánto le tarda a la luz en llegar a la luna desde la tierra.

**Guardando el Workspace o área de trabajo**: Utilizamos esto si estamos trabajando con muchas variables y datos y queremos retomar nuestro trabajo más adelante.

Digitar el comando “**save**” en la ventana de comando:

|  |  |
| --- | --- |
|  | El archivo quedará guardado en la carpeta actual de Matlab. |

También se puede guardar por el menú “**File**” y luego “**Save Workspace As**”

|  |  |
| --- | --- |
|  | De esta forma se puede seleccionar en que carpeta se guardará. |

**El efecto punto y coma y puntos suspensivos …**

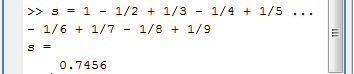
En Matlab se utiliza el “ ; “ al final de una línea para indicar que el comando terminó, además evita la confirmación de la ejecución de una asignación, también llamado eco, o sea, no reescribe el valor en la ventana de comando. P.E:

|  |  |
| --- | --- |
| Comando sin ; | Comando con ; |

El punto y coma sirve también para separar expresiones e ingresar varias líneas al mismo tiempo, P.E:

|  |  |
| --- | --- |
|  |  |
| Ingresa el valor de ambas variables sin hacer eco. | Ingresa el valor de 3 variables, pero sólo dos tienen punto y coma, por lo que hace eco, o muestra el valor de la variable y. |

Se utilizan puntos suspensivos “ ... “ o ellipsis (nombre en inglés), cuando necesitamos continuar una expresión en la siguiente línea, ya sea porque es muy larga o nos quedamos sin espacio:



**Mensajes de respuesta en Matlab**

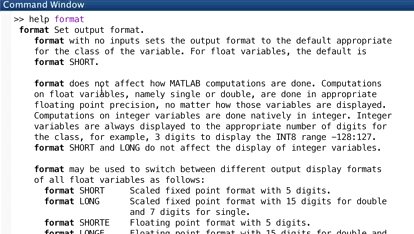
**inf**: Cuando el resultado de una operación es infinito.

**NaN**: significa not a number, por ejemplo el resultado de la operación 0/0

**Cómo usar la ayuda en Matlab:**

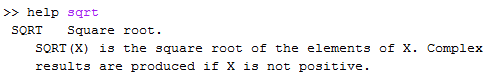
Matlab tiene una ayuda mediante la ventana de comando que se llama “ help ”, al digitar solamente esa palabra aparecerán todos los comandos por los cuales se puede consultar, o se puede solicitar ayuda específica sobre un comando si se tiene el nombre.

P.E 1: consulta ayuda acerca del comando format (visto anteriormente):



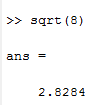
Aparecerá la documentación correspondiente a dicho comando junto con todas sus variaciones.

P.E 2: se desea calcular la raíz cuadrada de 8, el comando para solicitar ayuda sería help sqrt



Todas las ayuda de Matlab están en inglés (les tocó aprender o conseguir traductor).

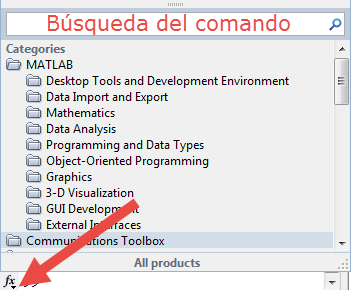
Pero observando con detenimiento la ayuda, podemos concluir que el número al que queremos aplicarle la raíz debe estar entre las comillas, por lo que el comando completo sería:



Otra forma de consultar la ayuda de Matlab es utilizando el comando “**doc**”, P.E:

|  |  |
| --- | --- |
|  | De esta forma aparecerá la ayuda en una ventana nueva y no en la ventana de comando como con **help**. |

Otra forma de solicitar ayuda a Matlab es dando clic en el botón de Fx en la ventana de comando:



La mejor forma de aplicar la ayuda es creando ejemplos, o consultar la ayuda online en la página <https://www.mathworks.com/help/> allí se encontrarán ejemplos aplicables a las diferentes versiones de Matlab a partir de la 2010.

**Archivos M-file o .M**

Matlab tiene varios tipos de archivos, uno de ellos es en el cual podemos digitar comandos propios junto con comandos de entrada y salida, se llaman scripts o archivos .m

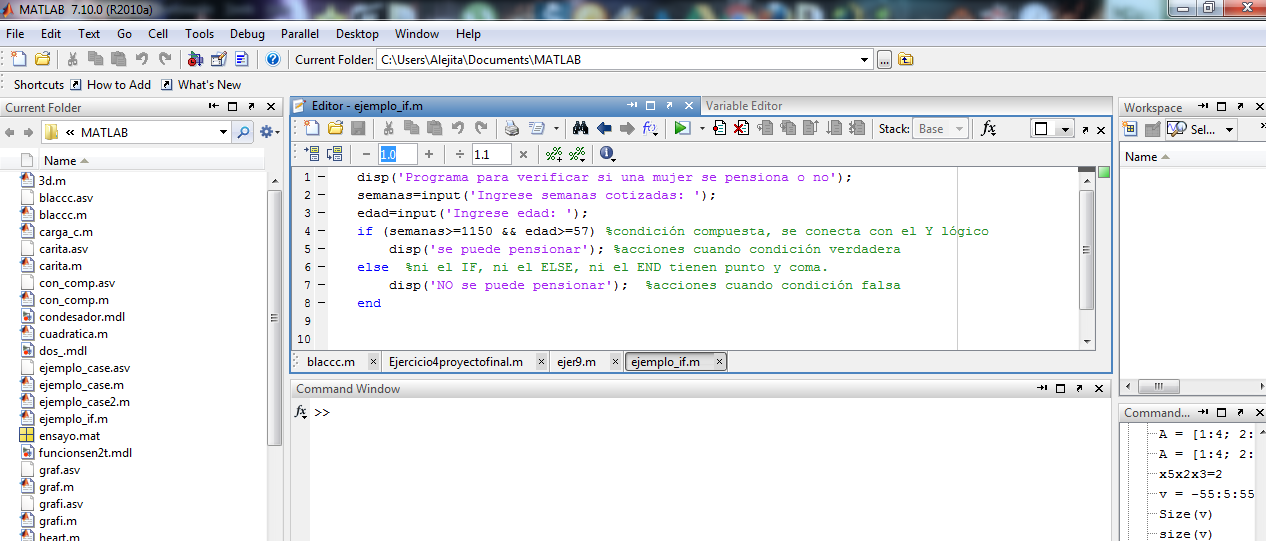
Para abrir el editor podemos dar clic en:

|  |  |
| --- | --- |
|  |  |

Y se puede anclar dicha ventana dentro del desktop de Matlab al dar clic en esta flecha:

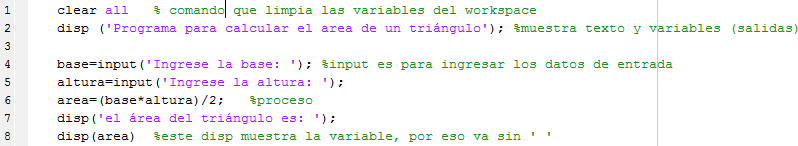


Se tendrá un entorno parecido al mostrado a continuación, el cual es útil para ejecutar programas y ver sus resultados sin necesidad de cambiar de ventana.



**Actividad 4**

Digitar el siguiente código en el editor de Matlab, (incluir los comentarios)



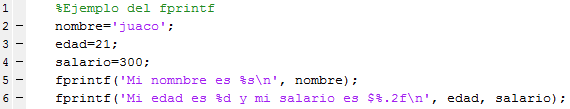
El comando **Clear all** limpia las variables del workspace, se aconseja comenzar todos los script o programas con este comando.

El comando **disp** puede mostrar texto y variables, si lo que va entre paréntesis lleva ‘ ’ (comillas sencillas) mostrará dicho texto, pero si se desea mostrar una variable debe ir sin comillas. P.E:



**Otra forma de mostrar mensajes en Matlab (fprintf)**

El comando **disp** muestra texto y variables de forma separada, pero existe un comando, (originario del lenguaje C), que le permite mostrar, en la misma línea, texto y variables. P.E:



En la línea 5 se muestra **%s** que sirve para mostrar variables de tipo cadena de caracteres o textos,

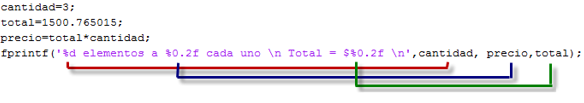
En la línea 6 aparece el **%d** que es para mostrar variables de tipo entero y **%f** para variables de tipo flotante o decimal, el **0.2** es para indicar de cuántos dígitos se mostrará el decimal.

Y para separar la línea en el programa (salto de línea), se utiliza el **\n**.

Resultado en Matlab:



Se pueden enviar varias variables en una sola línea como se muestra en el ejemplo:



En las líneas se observa que:

* **%d** sirve para mostrar el valor que tiene la variable **cantidad**, no tiene restricción ni formato, también indica que es una variable entera.
* **%0.2f** indica que la variable a la que corresponde, en este caso **precio**, va a tener dos decimales de precisión, y que la variable es de tipo decimal o flotante.
* **%0.2f** para la variable total, presenta un formato similar al anterior, indicando que sólo se mostrará el resultado con dos dígitos, sin importar que el dato original sea más grande, (observar línea de código).

El resultado en la ventana de comando será este:



**En resumen:**

%s – Muestra una cadena de caracteres o texto.

%c – Muestra un caracteres solo (solo una letra).

%d – Número entero.

%f – Número decimal o flotante.

\n – Abre una nueva línea (va a la siguiente línea para seguir imprimiendo).

**El uso del punto y coma en Matlab:**

La mayoría de las líneas en los scripts de Matlab llevan punto y coma (**;)** para indicar el final de la línea, esto se utiliza para todas, menos para las que contengan comandos de programación (Control Flow) como son If, Else, End, For, While, Do While.

Cuando Matlab detecta que no se ha puesto un punto y coma, aparece un subrayado rojo con un fondo amarillo, esto significa que se ha detectado un error de sintaxis.

|  |  |
| --- | --- |
|  |  |

Las líneas concernientes a If no llevan punto y coma, pero no aparecerá error, porque Matlab la confundirá con una línea normal y no la ejecutará de forma correcta, en otras palabras, este tipo de error Matlab no lo mostrará, el programador deberá darse cuenta en la ejecución del programa, ya que podría no obtener los resultados esperados.

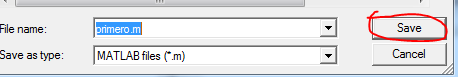


**Para ejecutar un archivo**

Primero se debe guardar dando clic en el icono de diskette en la parte superior del editor.

**Para guardar los archivos de Matlab se deben tener en cuenta algunas recomendaciones:**

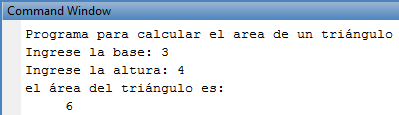
* Los nombres de los archivos NO de deben tener espacios al principio, intermedios o al final.
* Tampoco empezar el nombre con un número, pueden tener números al intermedio del nombre o al final. P.E: 1area (incorrecto), area1triang (correcto).
* No se deben llamar igual que las funciones de Matlab, P.E: existe una función que se llama sqrt (para calcular la raíz cuadrada), nuestros archivos .M no podrán llevar ese mismo nombre.
* No deben llamarse igual que una variable utilizada en el mismo archivo. P.E: si dentro del programa existe una variable que se llama **area**, el programa no podrá llamarse igual.
* Se aconseja que los nombres se escriban en minúsculas.



Al guardar el archivo se da clic en ejecutar o se puede presionar la techa F5 en la parte superior del teclado.



En la ventana de comando se ejecutará el programa:



Nótese que los comentarios (texto que va después del signo %), no se muestran en la ejecución, ya que es una ayuda para el programador, los comentarios sirven para explicar o tomar nota acerca de cómo funcionan algunas líneas del código, son útiles como referencias en el futuro.