

**BASADO EN MASTERING MATLAB 7 POR  
HANSELMAN Y LITTLEFIELD**

# INTRODUCCION A MATLAB

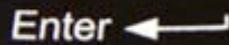
BASADO EN MASTERING MATLAB 7 POR  
HANSELMAN Y LITTLEFIELD

# CARACTERISTICAS BASICAS

En principio las funciones basicas de Matlab funcionan al estilo de una calculadora basica en la computadora, ejemplo:

Sumar dos numeros

```
>>2+2
```



```
ans=
```

```
4
```

Por lo tanto las operaciones basicas son:

Suma	+	2+3
Resta	-	9-6
Multiplicacion	*	3*4
Division	/ o \	7/19 o 19\7
Potencia	^	4^5

# CARACTERISTICAS BASICAS

Creacion de variables:

Para asignar un valor a una variable se debe poner el nombre de la variable el signo = y el valor que se quiere asignar, ejemplo:

Si queremos atribuirle 6 a la variable “nombre” entonces

```
>>nombre = 6
```



```
nombre =
```

```
6
```

Matlab reconoce mayusculas y minusculas, por lo que nombres como “Costo”, “COsto”, “COSTO”, “CostO”, son variables distintas para Matlab.

# ALGUNAS VARIABLES IMPORTANTES

Variables Especiales	Descripción
<b>ans</b>	Por defecto nombre de variable usada para resultados
<b>pi</b>	Radio de la circunferencia de un círculo al diámetro (~3.1416)
<b>i ó j</b>	$\sqrt{-1}$
<b>inf</b>	Tendencia para infinito (ejemplo, 1/0)
<b>NaN o nan</b>	No es un número (ejemplo, 0/0)

Nota: No es apropiado utilizar los nombres de estas variables para modificarlas, pues cualquier cambio puede generar errores.

# ALGUNAS FUNCIONES IMPORTANTES

Funciones Trigonométricas	Descripción
<b>cos</b>	Coseno
<b>cot</b>	Cotangente
<b>csc</b>	Cosecante
<b>sec</b>	Secante
<b>sin</b>	Seno
<b>tan</b>	Tangente
<b>cosh</b>	Coseno hiperbólico
<b>coth</b>	Cotangente hiperbólico
<b>csch</b>	Cosecante hiperbólico
<b>sech</b>	Secante hiperbólico
<b>sinh</b>	Seno hiperbólico
<b>tanh</b>	Tangente hiperbólico
<b>acosh</b>	Inversa de coseno hiperbólico
<b>acot</b>	Inversa de cotangente
<b>acsc</b>	Inversa de cosecante
<b>acsch</b>	Inversa de de cosecante hiperbólico
<b>a"nombre"</b>	Inversa de la función nombre
<b>a"nombre"h</b>	Inversa de la función nombre hiperbolica

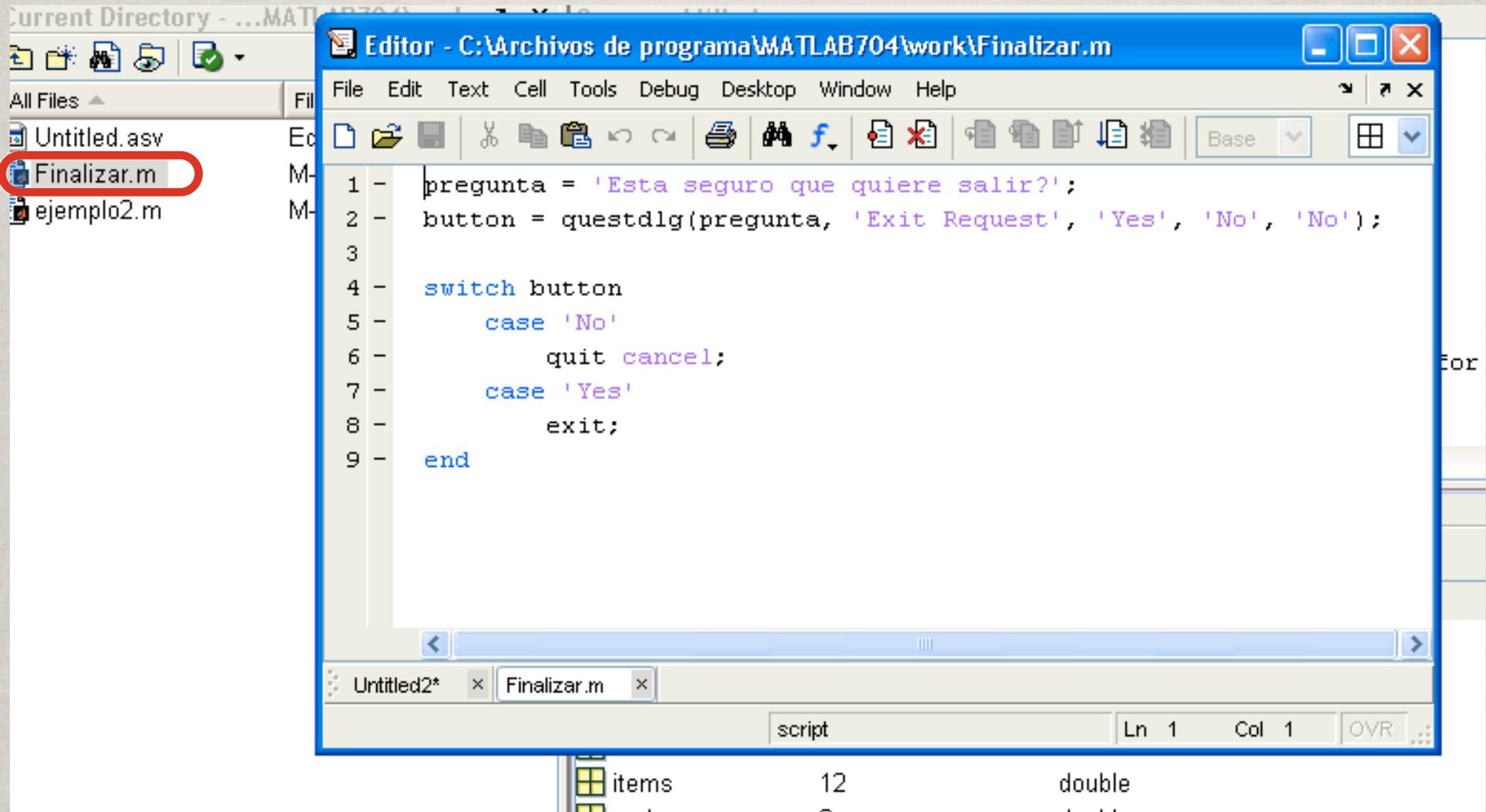
# ALGUNAS FUNCIONES IMPORTANTES

Funciones Exponenciales	Descripción
<b>^</b>	Potenciación (ej., $4^2 = 16$ )
<b>exp</b>	Función exponencial
<b>expm1</b>	Exponencial menos 1 (es decir: $\exp(x) - 1$ )
<b>log</b>	Logaritmo Natural
<b>log10</b>	Logaritmo Natural en Base 10
<b>log1p</b>	Logaritmo Natural de $(x+1)$ (es decir: $\log(x+1)$ )
<b>log2</b>	Logaritmo natural en base 2
<b>nthroot</b>	n-esima raiz de numeros reales
<b>sqrt</b>	Raiz Cuadrada
Funciones Complejas	Descripción
<b>abs</b>	Valor Absoluto o Magnitud
<b>angle</b>	Angulo en radianes
<b>conj</b>	Conjugación Compleja
<b>imag</b>	Parte imaginaria
<b>real</b>	Parte real
<b>isreal</b>	'Verdad' para valores reales

# ALGUNAS FUNCIONES IMPORTANTES

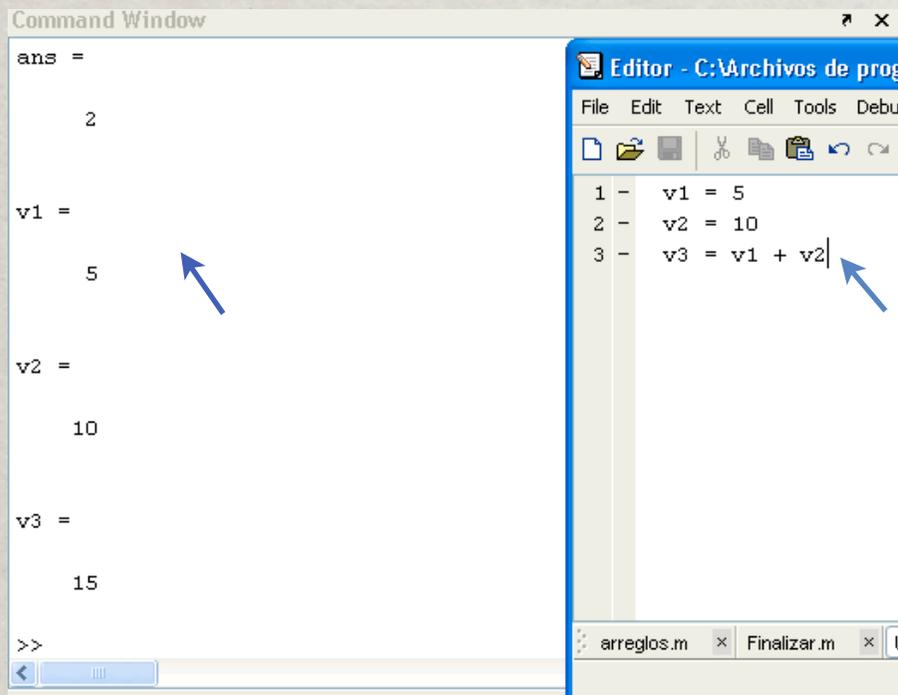
Funciones Especializadas	Descripción
<b>besselj</b>	Primera clase de Funcion Bessel
<b>bessely</b>	Segunda clase de Funcion Bessel
<b>besselh</b>	Tercera clase de Funcion Bessel
<b>beta</b>	Funcion Beta
<b>ellipj</b>	Funcion eliptoca de Jacobi
<b>ellipke</b>	Integral eliptica completa
<b>erf</b>	Funcion Error
<b>gamma</b>	Funcion Gamma
<b>legendre</b>	Funcion Asociada de Legendre
<b>cross</b>	Producto cruz de vectores
<b>dot</b>	Producto punto de vectores

# ARCHIVOS .M

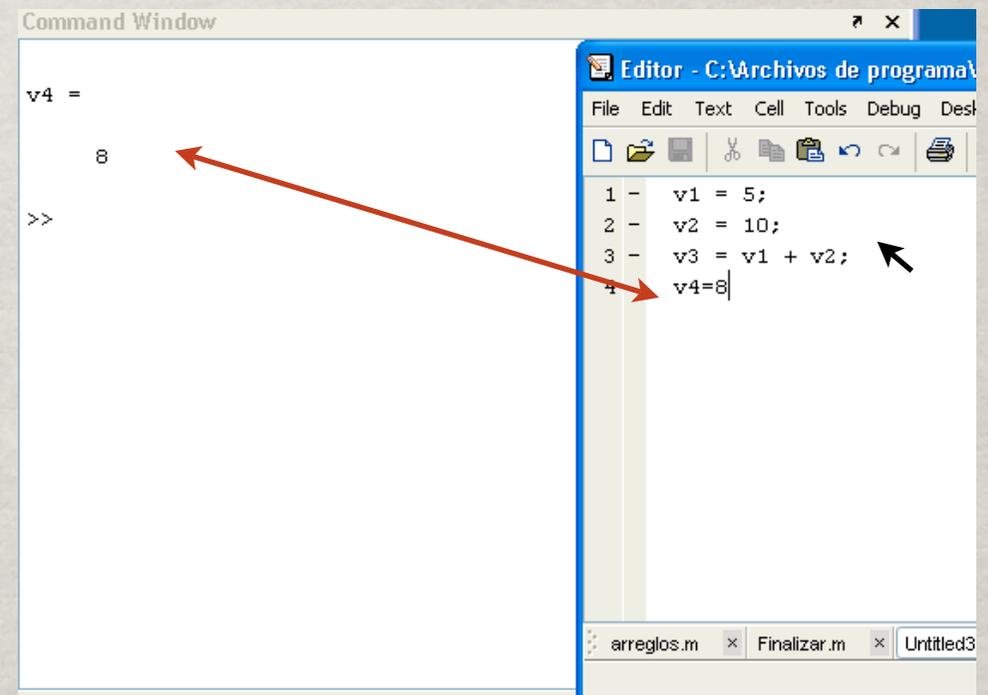


# ARCHIVOS .M

Si no queremos ver el resultado de una variable o función en tiempo de compilación ponemos punto y coma ‘;’ al final de la línea, ejemplo:



The screenshot shows the MATLAB Command Window on the left and the Editor on the right. The Command Window displays the results of variable assignments: `ans = 2`, `v1 = 5`, `v2 = 10`, and `v3 = 15`. The Editor shows the corresponding code in `arreglos.m`:  
1 - `v1 = 5`  
2 - `v2 = 10`  
3 - `v3 = v1 + v2`  
Blue arrows point from the code in the Editor to the output in the Command Window.



The screenshot shows the MATLAB Command Window on the left and the Editor on the right. The Command Window displays the results of variable assignments: `v4 = 8`. The Editor shows the corresponding code in `arreglos.m`:  
1 - `v1 = 5;`  
2 - `v2 = 10;`  
3 - `v3 = v1 + v2;`  
4 - `v4=8`  
A red arrow points from the code in the Editor to the output in the Command Window.

# ARREGLOS

Técnicas de Construcción de Arreglos	Descripción
<code>x=[2 2*pi sqrt(2) 2-3j]</code>	Crea vector columna conteniendo elementos arbitrarios.
<code>x=inicio:fin</code>	Crea un vector columna con la primera entrada 'inicio' y llenando sumando 1 hasta llegar a fin.
<code>x=inicio:medio:fin</code>	Crea un vector columna comenzando con inicio, continuando con medio, y terminando en fin.
<code>x=linspace(inicio, fin, n)</code>	Crea un vector columna comenzando con inicio y finalizando con fin de n elementos

Para transponer un vector utilizamos “'” ejemplo:

```
>> x=1:5  
  
x =  
  
    1    2    3    4    5  
  
>> z=x'  
  
z =  
  
    1  
    2  
    3  
    4  
    5
```

# OPERACIONES DE ARREGLOS

Operacion Elemento por Elemento	Descripcion $A=[a_1 a_2 \dots a_n]$ , $B=[b_1 b_2 \dots b_n]$ $c = \langle \text{escalar} \rangle$
<b>Adicion Escalar</b>	$A+c = [a_1+c a_2+c \dots a_n+c]$
<b>Substraccion Escalar</b>	$A-c = [a_1-c a_2-c \dots a_n-c]$
<b>Multiplicacion Escalar</b>	$A*c = [a_1*c a_2*c \dots a_n*c]$
<b>Division Escalar</b>	$A/c = c \setminus A = [a_1/c a_2/c \dots a_n/c]$
<b>Suma de Arreglos</b>	$A+B = [a_1+b_1 a_2+b_2 \dots a_n+b_n]$
<b>Multiplicacion de Arreglos</b>	$A.*B = [a_1*b_1 a_2*b_2 \dots a_n*b_n]$
<b>Division de Arreglos por derecha</b>	$A./B = [a_1/b_1 a_2/b_2 \dots a_n/b_n]$
<b>Division de Arreglos por izquierda</b>	$A.\setminus B = [a_1 \setminus b_1 a_2 \setminus b_2 \dots a_n \setminus b_n]$
<b>Potenciacion de Arreglos</b>	$A.^c = [a_1^c a_2^c \dots a_n^c]$
	$c.^A = [c^{a_1} c^{a_2} \dots c^{a_n}]$
	$A.^B = [a_1^{b_1} a_2^{b_2} \dots a_n^{b_n}]$

# MATRICES (ARREGLOS DE ARREGLOS)

La forma usual de crear una matriz de  $m \times n$  es la siguiente:

$$A = [a_{11} \ a_{12} \ \dots \ a_{1n}; \ a_{21} \ a_{22} \ \dots \ a_{2n}; \ \dots ; \ a_{m1} \ a_{m2} \ \dots \ a_{mn}]$$

donde  $A(i,j) = a_{ij}$ .

Direcciones de los Arreglos	Descripción
$A(r,:)$	Dirección de un subarreglo dentro de A definido por el vector del índice de filas deseadas en r y todas las columnas
$A(:,c)$	Dirección de un subarreglo dentro de A definido por todas las columnas y el vector índice deseado en columnas c
$A(:)$	Dirección de todos los elementos de A como vectores columnas tomando columna por columna
$A(k)$	Dirección de un subarreglo dentro de A definido por el único vector índice k.

# TAMAÑO DE LOS ARREGLOS

Para A Matriz:

Tamaño de los arregls	Descripción
<b>s=size(A)</b>	Retorna un vector columna s cuyo primer elemento es el numero de filas en A y el segundo el numero de columnas
<b>[r,c]=size(A)</b>	Retorna 2 escalares r, c, conteniendo el numero de filas y columnas respectivamente
<b>r=size(A,1)</b>	Retorna el numero de filas en A
<b>c=size(A,2)</b>	Retorna el numero de columnas de A
<b>n=numel(A)</b>	Retorna el numero total de elementos en A.

# CONTROL DE FLUJO

## Operadores

Operadores de relacion	Descripción
<	Menor que
<=	Menor o igual que
>	Mayor que
>=	Mayor o igual que
"=="	Igual a
"!="	Distinto a

Operadores logicos	Descripción
&	Elemento por elemento "AND" para arreglos
	Elemento por elemento "OR" para arreglos
~	NO (Negacion de una sentencia)
&&	"AND" para escalares
	"OR" para escalares