انواع المتغيرات في برنامج MATLAB

1. متغيرات مسبقة التعريف في البرنامج Predefined Variables

هي مجموعة من الثوابت constants والقيم الخاصة special values المحجوزة في البرنامج حيث تأتي معرفة تلقائيا في بنية البرنامج الداخلية ويمكن استخدامها مباشرة دون ان يتم تعريفها. الجدول الاتي يوضح امثلة من هذه المتغيرات.

ans	هو المتغير الافتراضي لأي ناتج عملية حسابية في برنامج MATLAB عند عدم اعطاء اسم متغير للقيمة الناتجة، مثلا >> 5+3 ans = 8 يقوم البرنامج تلقائيا بخزن ناتج عملية الجمع في المتغير ans لأننا لم نعرف اسم متغير لناتج هذه العملية الحسابية .
pi	هي النسبة الثابتة 22/7= π وتعرف في البرنامج على الشكل التالي : >> pi ans = 3.1416
Inf	يعبر عن قيم اللانهاية infinity ∞ الناتجة من القسمة على صفر >> 1/0 ans = Inf
NaN	تعبر عن القيمة التي ليست رقم ، وهي اختصار جملة Not a Number وقد تنتج عندما تكون قيمة الناتج يساوي (0/0) او لتعبر عن ان المعلومات مفقودة ، او غير متوفرة ، والذي قد يكون سببه فشل البرنامج في الحساب. >> 0/0 Warning: Divided by zero. ans = NaN
i , j	يتم استخدام احد هذين الرمزين عند تعريف الاعداد المركبة (المعقدة) فهما يمثلان الجزء التخيلي للاعداد المركبة حيث يتم استخدامهم على الشكل التالي : >> 3+4*i ans = 3.0000 + 4.0000i >> 3+4*j ans = 3.0000 + 4.0000i مع ملاحظة ان كلا الرمزين يمثلان العدد المركب $\sqrt{-1}$

2. متغيرات تعرف بواسطة المستخدم User- defined Variables

وهي المتغيرات التي يقوم المستخدم بتعريفها بإعطائها قيمة عددية او نصية ،وسيتعرف البرنامج على نوع هذه المتغيرات دون تحديده كما ذكرنا سابقا ، ويتم تسمية المتغير في برنامج MATLAB ضمن شروط معينة.

شروط تسمية المتغيرات داخل برنامج Matlab:-

1. يجب ان يبدأ اسم المتغير بحرف وليس برقم او برمز فمثلا لا يمكن كتابة اسم المتغير على الشكل 1a= 5 وبدلا من ذلك يمكننا كتابة اسم المتغير على الشكل a1=5 .
2. لا يمكن ان يحتوي اسم المتغير على مسافة (فراغ)، فمثلا لا يمكن كتابة اسم المتغير على الشكل a val وبدلا من ذلك يمكن استخدام علامة الشرطة السفلية (_) Underscore على الشكل a_val.
3. يجب ان لايحتوي اسم المتغير على بعض الرموز الخاصة مثل ,-, +, *, %, ?, ^, @, # , <, > , ( ), [ ], ! , \,/ .
4. يجب ان لا يأخذ اسم المتغير اسم امر او دالة محجوزة في برنامج MATLAB ،فمثلا لا يمكن تسمية المتغير if لان هذا الاسم من الكلمات المحجوزة reserved words او الكلمات المفتاحية keywords داخل اللغة، ولكن يمكن استخدام

كلمات شبيهة لها من خلال دمج ارقام معها مثل if1 او جعل اول حرف منها كبيرا capital مثل IF .وهذه بعض الكلمات المحجوزة داخل البرنامج if elseif else end for while break continue return switch try catch function global persistent case otherwise

يتم التعرف على قائمة الكلمات المحجوزة في برنامج MATLAB بكتابة الامر iskeyword في نافذة الاوامرcommand window كما يلي:

```
>> iskeyword
ans =       'break'
  'case'
  'catch'
  'classdef'
  'continue'
  'else'
  'elseif'
  'end'
  'for'
  'function'
  'global'
  'if'
  'otherwise'
  'parfor'
  'persistent'
  'return'
  'spmd'
  'switch'
  'try'
  'while'
```

5. يجب ان لا يزيد عدد الاحرف التي يتكون منها اسم المتغير عن 63 حرف وسيهمل اي رمز يزيد عن 63 حرف.

6. سبق وذكرنا برنامج MATLAB حساس لحالة الاحرف case sensitive حيث يميز بين الاحرف الكبيرة capital letters والاحرف الصغيرة small letters ،فمثلا عند القيام بتسمية متغير بالاسم a فان برنامج MATLAB يتعامل معه على ان له قيمة مختلف عن المتغير A .

ملفات M النصية

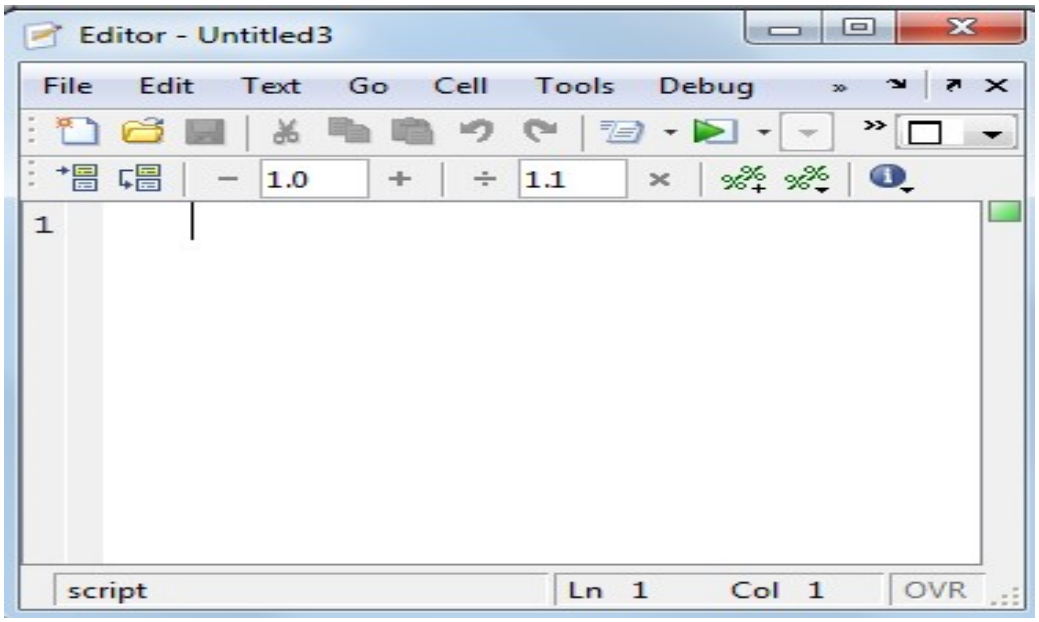
Script M – Files

ملفات M – Files:- هي نوع من انواع الملفات النصية Script Files التي يعمل عليها برنامج MATLAB كوسيلة لإدخال الاوامر والرموز البرمجية ، يسمى هذا الملف "M-File" (

انشاء ملف M – File جديد:-

لإنشاء ملف نصي جديد لكتابة برنامج MATLAB وفي أي اصدار قديم او جديد نبحث عن New M-File أو New Script

وباتباع احدى الطرق سوف تظهر نافذة جديدة ، تأخذ الشكل التالي:



سوف يحدد برنامج MATLAB اسما افتراضيا لهذا الملف هو Untitled وعند حفظ هذا الملف يعمل برنامج MATLAB على اضافة الامتداد (m .*) الى اسم هذا الملف .

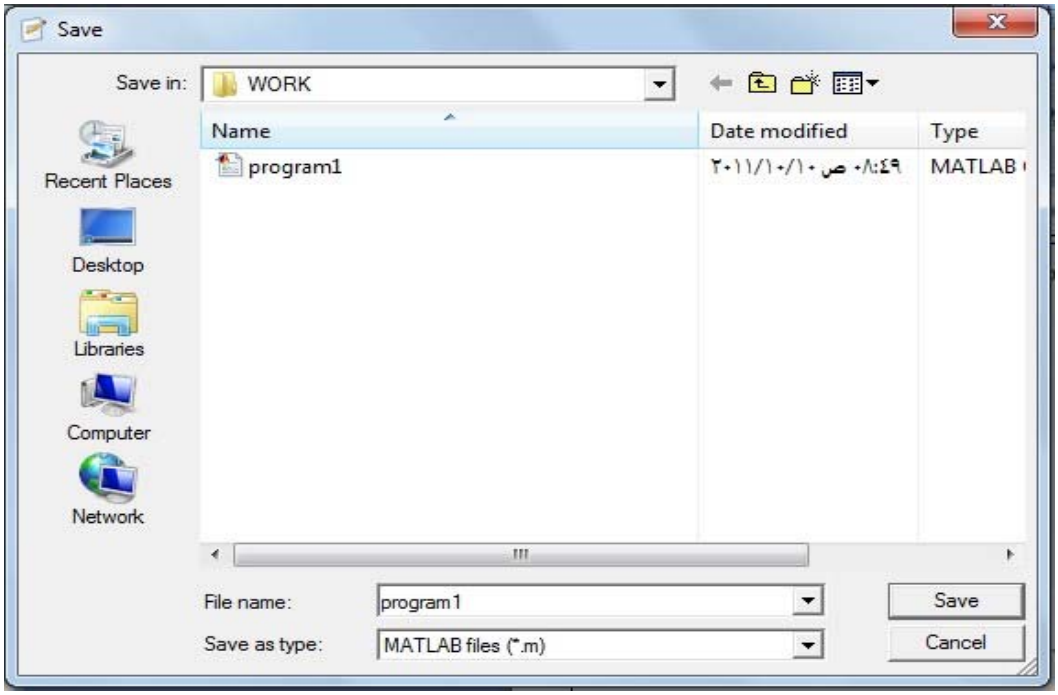
شروط حفظ ملف M-File :-

1. يجب ان يبدأ اسم الملف بحرف وليس برقم او برمز فمثلا لا يمكن كتابة اسم الملف على الشكل 1test.m وبدلا من ذلك يمكننا اسم الملف على الشكل test1.m.
2. يجب ان لا يسمى اسم الملف على اسم امرا معروفا او دالة مبنية داخل برنامج MATLAB فمثلا لا يمكن تسمية الملف بالكلمة if لان هذا الاسم يمثل احدى الدوال الداخلية للبرنامج Built in functions
3. يجب ان لا يحتوي اسم الملف على مسافات فاصلة Space فمثلا لا يمكن كتابة اسم الملف على الشكل التالي test a وبدلا من ذلك يمكن استخدام علامة الشارحة التحتية under score (_)على الشكل التالي test_a .
4. يجب ان لا يحتوي اسم الملف على بعض الرموز الخاصة مثل , * \ ، / ، ! ، ؟ باستثناء علامة الشارحة التحتية (_) under score حيث يمكن استخدامها سابقا

حفظ ملف M – File :-

يحفظ ملف برنامج MATLAB من أمر Save الموجود في Editor او اضغط على مفتاحي Ctrl+S من لوحة المفاتيح keyboard

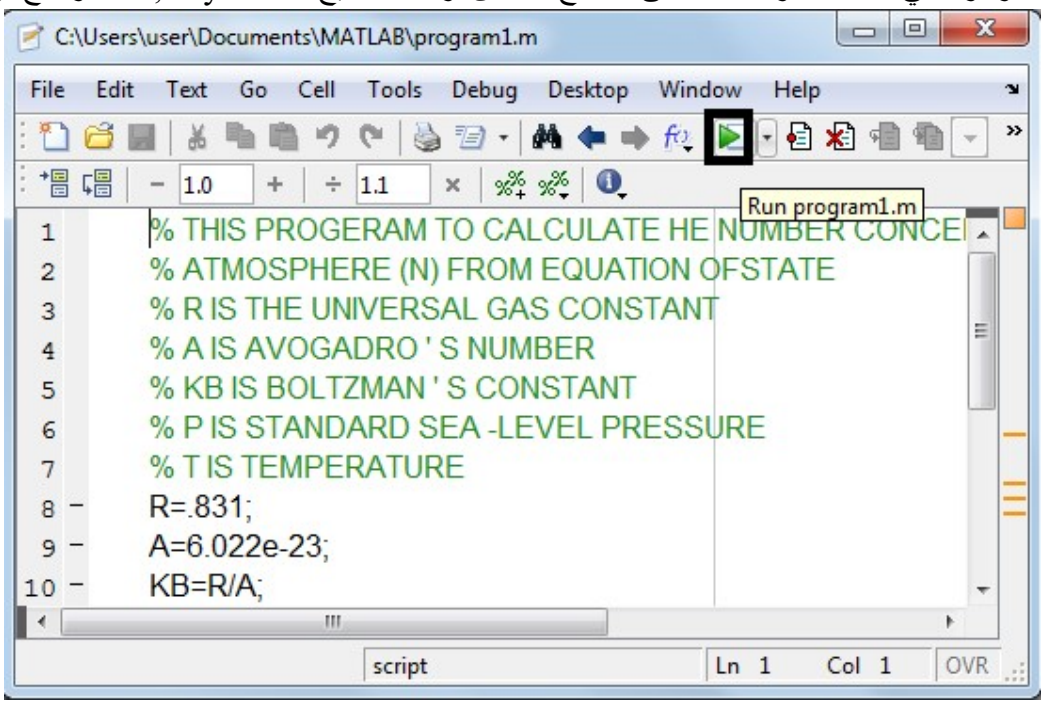
وعند حفظ ملف M – File المتضمن كود البرنامج باستخدام اي من الطرق السابقة يظهر مربع الحوار Save file as لاختيار مسار حفظ الملف M – File ، كما في الشكل التالي: -



اكتب اسم الملف المراد حفظه في حقل File name ، مع مراعاة شروط اختيار اسم الملف السابق ذكرها ، كما نلاحظ ان الملف المحفوظ يأخذ الامتداد (m . *) .

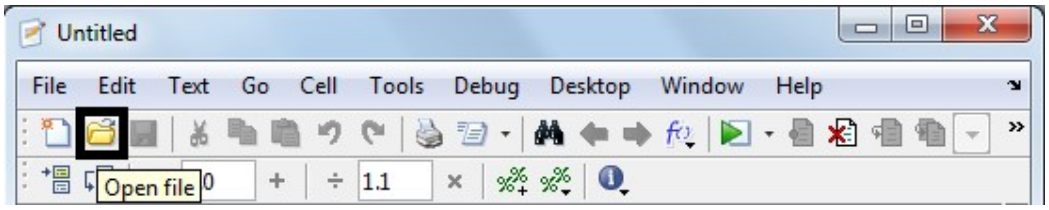
- <u>تنفيذ البرنامج:</u>

يتم تنفيذ برنامج MATLAB المكتوب داخل ملف M – File بالضغط على ايقونة زر التشغيل Run الموجودة في Editor أوالضغط على المفتاح F5من لوحة المفاتيح Keyboard, كما موضح في الشكل التالي:

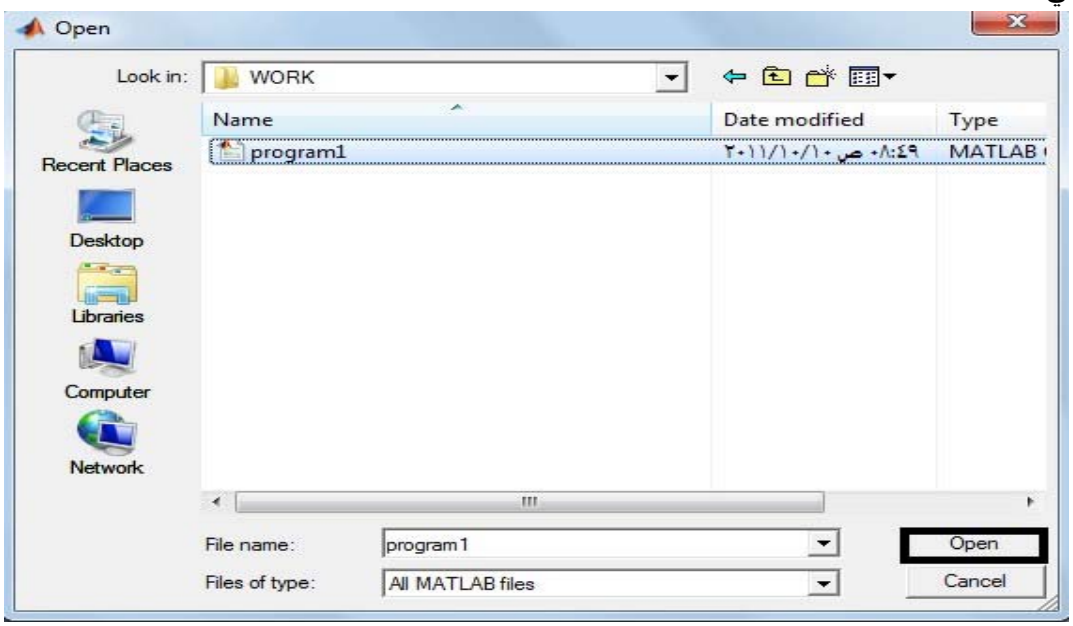


فتح ملف M – File سبق حفظه :-

لفتح ملف برنامج M – File سبق حفظه: - اختر امر الفتح Open من نافذة سطح مكتب برنامج MATLAB او من نافذة منفتح البرامج Editor او بالضغط على مفتاحي Ctrl+O من لوحة المفاتيح Keyboard ، كما في الشكل التالي: -



سوف يظهر لك مربع الحوار Open لاختيار اسم الملف المراد فتحه ثم اضغط على مفتاح Openكما في الشكل التالي: -



العمليات والدوال الرياضية

❖ **اسبقيات(أولوية) العمليات الرياضية في برنامج MATLAB :-**

1- الاقواس.
2- الرفع الى القوة.
3- الضرب والقسمة.
4- الجمع والطرح.

ملاحظة: - تحسب العمليات الرياضية من اليسار الى اليمين التي تحتوي على نفس الاسبقية.

Ex:- >>(3 + 22) * (15.7 – 8)

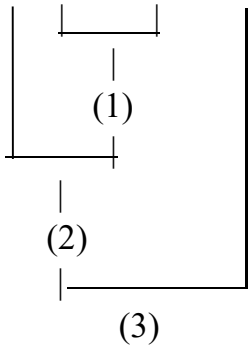
(1) (2)

(3)

ans= 192.5

Ex:-

>>3 + 22 * 15.7 – 8



ans= 340.4

❖ **ملاحظات** :-

1. عند وضع علامة الفاصلة المنقوطة (;) semicolon في نهاية اي من الاوامر) المدخلات (السابقة فسوف يتم تنفيذ الامر دون اظهار الناتج . لاحظ الفرق في المثال التالي:

```
>> a=4+5
a =     9
>> a=4+5;
```

2. يمكن استدعاء متغير ومعرفة قيمته بعد ان تم ادخاله في البرنامج وذلك من خلال كتابة المتغير فقط والضغط على مفتاح enter كما يلي

```
>> a
a =     5
```

بعض دوال التحكم في نافذتي command window, workspace

يمكن تلخيص دوال التحكم في نافذتي command window , workspace وتنظيمها في الجدول التالي:

Operation	Function
مسح جميع محتويات نافذة command window فقط دون مسحها من نافذة workspace	clc
مسح جميع محتويات نافذة workspace المتضمنة جميع المتغيرات التي تم استخدامها في البرنامج	clear
مسح المتغيرات a b c فقط من نافذة workspace	clear a b c
مسح جميع المتغيرات الموجودة في نافذة workspace والتي تبدأ بحرف a	clear a*
عرض محتويات نافذة workspace والمتضمنة جميع المتغيرات التي تم استخدامها في البرنامج	who
عرض محتويات نافذة workspace بالتفصيل (الاسم ،الابعاد ، الحجم ،النوع)	whos
حفظ جميع محتويات نافذة workspace في الملف الافتراضي matlab.mat	save

الدوال الرياضية :

Example	Function in MATLAB form	Operation
>> exp(0) ans = 1	exp(x)	الدالة الاسية
>> log(1) ans = 0	log(x)	دالة اللوغارتيم الطبيعي ln

>> log10(2) ans = 0.3010	log10(x)	دالة اللوغارتيم للاساس10
>> log2(2) ans = 1	log2(x)	دالة اللوغارتيم للاساس2
>> pow2(3) ans = 8	pow2(x)	دالة الرفع الى قوة للاساس2
>> sqrt(4) ans = 2	sqrt(x)	دالة الجذر التربيعي
>>power(3,3) ans = 27	power(x,y)	دالة الرفع للاساس x
>> factorial(3) ans = 6	factorial(x)	دالة المضروب
>> abs(-3) ans =3	abs(x)	دالة المطلق

الدوال المثلثية Trigonometric functions:

تكتب الدوال المثلثية كما هي

sin(angle), cos(angle), tan(angle), sec(angle), csc(angle), cot(angle)

مثلا:

>> sin(5) ans = -0.9589
>> csc(5) ans = -1.0428

اما معكوس الدوال المثلثية فيكون كالاتي:

>> asin(5) ans = 1.5708 - 2.2924i	asin(x)	معكوس دالة sin
>> acos(5) ans = 0 + 2.2924i	acos(x)	معكوس دالة cos
>> atan(5) ans = 1.3734	atan(x)	معكوس دالة tan
>> asec(5) ans = 1.3694	asec(x)	معكوس دالة sec
>> acsc(5) ans = 0.2014	acsc(x)	معكوس دالة csc
>> acot(5) ans = 0.1974	acot(x)	معكوس الدالة cot

ملاحظة: **- جميع الدوال المثلثية اعلاه مقاسة بالتقدير الدائري Radian degree ، ولغرض تحويل التقدير الى الدرجات degrees فانه يمكن ذلك كالاتي :**

بإضافة الحرف d قبل اي دالة ،مثلا نكتب دالة sin بالشكل sind.

أو يتم تحويل الزاوية في برنامج MATLAB باستخدام احد الامرين التاليين :-

1- **rad2deg :-** يستخدم هذا الامر لتحويل الزاوية نصف قطرية (radians) الى الدرجات (degrees) .

2- **deg2rad :-** يستخدم هذا الامر لتحويل الزاوية من الدرجات (degrees) الى القياس نصف

دوال التدوير والبقية Routing & Remainder functions

1-fix :- Round toward zero

دالة التقريب الى اقرب رقم صحيح اهمال الجزء الكسري

Ex:-

```
>>A = [-1.9 , -0.2 , 3.4 , 2.4 + 3.6] A =
  -1.9000      -0.2000      3.4000      6000
>>fix(A) ans =
  -1      0          3          6
```

2-round :- Round to nearest integer

دالة التقريب الى اقرب عدد صحيح.

Ex:-

```
>>round(A) ans =
 -2          0          3              6
```

3- **ceil** :- Round toward positive infinity

دالة التقريب باتجاه اللانهاية الموجبة (∞+) , دالة التقريب نحو اليمين.

Ex:-

```
>>ceil(A) ans =
 -1      0          4          6
```

4- **floor** :- Round toward negative infinity

دالة التقريب باتجاه اللانهاية السالبة (∞-), دالة التقريب نحو اليسار.

Ex:-

```
>>floor(A) ans =
  -2          -1          3          6
```

5- **rem** :- Remainder after division

دالة الباقي من القسمة. R = rem(X,Y)

Ex:-

```
>> rem(8,5) ans =
    3
>> rem(8,8)
ans =
    0
>> rem(8,10)
ans =
```

```
   8
>> rem(-1,2)
ans =
  -1
```

6- **mod** :- Modulus after division

دالة الباقي من القسمة. M = mod(X,Y)

```
Ex:-
>>mod(13,5) ans =
   3
Ex:-
>>rem(-5,2)
ans= -1
>>mod(-5,2)
ans=1
```

7-**sign :-** Signum function

دالة الاشارة: - اذا كان الرقم المرسل اكـبـــرمن الصفر فان ناتج الدالة 1
اذا كان الرقم المرسل اصغــرمن الصفر فان ناتج لدالة 1- اذا كان الرقم المرسل مساوي الى الصفر فان ناتج الدالة 0

```
Ex:-
>> sign(0) ans =
   0
>> sign(-9) ans =
  -1
>> sign(0.1) ans =
   1
```

اوامر الادخال والاخراج في برنامج MATLAB

Input I/P & Output O/P Commands in MATLAB

اوامر الادخال input(I/P) :

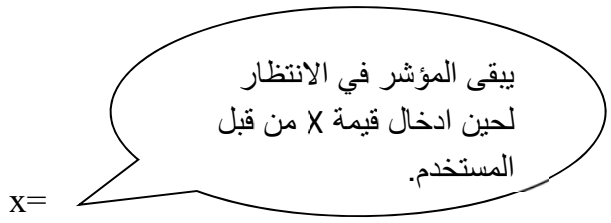
يطبع الامر input رسالة نصية للمستخدم على الشاشة كطلب إدخال بيانات عددية او حرفية وتعيينها الى متغير يعرفه المستخدم. ويستخدم الامر input على احدى الصورتين التاليتين :

اولا : ادخال بيانات عددية

```
x=input('displayed strings')
```

Displayed strings: هي مجموعة من الكلمات تمثل رسالة نصية يتم عرضها للمستخدم لتعبر عن القيمة التي سيقوم المستخدم بإدخالها ، x هو المتغير الذي يتم ادخال قيمته.

```
>> x=input('x=');
```



x =5

ملاحظة: تستخدم عبارة الادخال هذه عوضا عن الطريقة السابقة للإدخال المباشر للمتغيرات في برنامج MATLAB للتحكم بالقيم المعطاة عند كل تنفيذ للبرنامج، طريقة الادخال المباشر سوف تعطي قيمة ثابتة لا يمكن تغييرها ولكن باستخدام طريقة الادخال هذه يمكن اعطاء قيم اخرى.

مثال: لإدخال درجة الحرارة وعرض رسالة نصية لتدل على ان المتغير المدخل هو درجة الحرارة

```
>> T=input('Enter the temperature')
Enter the temperature
```

وبعد ظهور الجملة أعلاه قم بإدخال قيمة T الذي يمثل درجة الحرارة ولتكن 12

```
T =
   12
```

ثانيا : ادخال بيانات رمزية

```
X=input("displayed strings','s')
```

تستخدم هذه الصورة لاستقبال سلسلة حرفية يقوم المستخدم بإدخالها، حيث يستخدم الحرف 's' الذي يرمز للكلمة (string) ويفيد تحديد نوع البيانات في تحديد نطاق التخزين المستغل من الذاكرة المؤقتة للبرنامج مما يساعد على عدم اهدار الذاكرة المؤقتة للبرنامج.

مثال : نريد عرض رسالة نصية لتدل على ان درجة الحرارة مقاسة بالمقياس السيليزي

```
>> T=input('enter the measure of temperature ','s') enter the
measure of temperature
```

وبعد ظهور الجملة اعلاه قم بإدخال المقياس وليكن المقياس السيليزي Celsius تظهر

```
T =
'Celsius'
```

اوامر الاخراج : disp/display/fprintf (O/P)

تستخدم اوامر الاخراج لعرض قيم واسماء المتغيرات او التعبيرات النصية في نافذة محرر الاوامر Command window .

1. **الامر disp :** يستخدم الامر disp في عرض قيمة المتغير فقط سواء كانت عددية او نصية، ويستخدم هذا الامر على احدى الصورتين:

```
disp(x)
disp('displayed strings')
```

حيث يستخدم الامر الاول لعرض قيمة المتغيرx بينما يستخدم الامر الثاني لعرض تعبير نصي معين يتم ادخاله بين علامتي اقتباس مفردة .

مثال : لعرض قيمة عددية

```
>> x=100;
```

```
>> disp(x)
   100
```

لعرض تعبير نصي

```
>> disp(' the value of x is ')
the value of x is
```

2. **الامر display** : يستخدم الامر display في عرض اسم المتغير ثم قيمته سواء كانت رقمية او نصية ، ويستخدم على الصورة التالية:

```
display(x)
```

حيث يستخدم هذا الامر لعرض اسم المتغير x ثم قيمته على نافذة command window

ولفهم الفرق الواضح بين الصورتين السابقتين لأوامر الاخراج لاحظ المثالين التاليين:

```
>> x=5;
>> disp(x)                %display only variable value
    5
>> display(x)             %display variable name and value
x =     5
```

3. الامر fprintf (File Print Format):

❖في حالة طباعة تعبير نصي فان الامر يكتب بالصيغة التالية **fprinf('text')** مثال :

```
>> fprintf('the amount of precipitation is')
the amount of precipitation is
```

❖اما في حالة طباعة تعبير نصي وقيمة لمتغير فان الامر يكتب بالصيغة التالية **:-**

fprintf('format string', list of variable)

ونعني بكلمة format هنا تنسيق البيانات ، اما variable فهي القيمة العددية للمتغير.

جمل التحكم والشرط والتكرار

PROGRAM CONTROL, CONDITION AND LOOP STATEMENTS

الجمل الشرطية: تقسم جمل الشرط الى نوعين وهي: -

1- جملة اذا الشرطية (if – statement) 2- جملة التوزيع (switch – case)

1- جملة اذا الشرطية (if – statement) :- وتستخدم للتحكم بطريقة سير البرنامج اعتمادا على تحقق الشرط **(condition)** ، كما يلي: -

- الجملة (if / else) :- يستخدم الامر else كجزء من جملة الشرط if ، فاذا تحقق شرط جملة if يترتب عليه تحقيق النتائج التالية لجملة if الى ما قبل جملة else مباشرة والا تحقق النتائج ما بعد else ، ويأخذ الصورة التالية

ex:-
x=input('enter the number =') if x > 0
disp(' x is positive number ') else
disp('x is negative number ') end

- الجملة (if / elseif) :- الغرض من هذه الصورة من صور جملة الشرط if هو اختبار مدى تحقق اكثر من شرط ويأخذ الصيغة التالية: -

if condition 1 statement 1 elseif condition 2
statement 2 elseif condition (n-1) statement (n-1)
else statement (n) end

مثال / اكتب برنامج بلغة MATLAB لإدخال عدد ثم يبين اذا كان العدد موجب ام سالب او يساوي صفر ؟

```
x = input ('enter the number =');
if  x > 0
disp (' this number is positive ')
elseif  x < 0
disp ('this number is negative ')
else
disp('this number is equal 0 ')
end
```

ملاحظة: - يجب ان تحتوي الجمل الشرطية على احد ادوات المقارنة او الادوات المنطقية (Logical or Relational Operators) الموضحة في الجدول التالي: -

Logic Operators المعاملات المنطقية	Operation الوظيفة
<	اقل من
<=	اقل من او يساوي
>	اكبر من
>=	اكبر من او يساوي
==	يساوي
~=	لا يساوي
&	And

&&	Short-circuit And (scalars) للقيم العددية فقط
|	Or
||	Short-circuit or (scalars) للقيم العددية فقط
~	Not

2- جملة التوزيع (switch – case)

تستخدم للتحكم بطريقة سير البرنامج اعتمادا على قيمة المتغير المعطى لها والصيغة العامة لها: -

switch variable case value 1 Statement 1

case value2 Statement 2 case value n Statement n otherwise Statement end

يبدأ البرنامج اولا بقراءة قيمة المتغير (variable) ثم يبدأ بمقارنتها مع القيم (value) بعد عبارة case اذا كانت قيمة المتغير variable تنطبق مع اي من القيم value يتم تنفيذ الجملة statement بعدها وهكذا ، واذا لا تنطبق اي من القيم value مع قيمة المتغير variable سوف ينتقل الى العبارة otherwise ويتم تنفيذ الجملة بعدها.

مثال / اكتب برنامج بلغة MATLAB لحساب قيمة الاطوال بوحدات السنتمتر.

```
n = input ('n=');
 switch(rem(n,3))
    case 0
      m = 'no remainder'
    case 1
      m = 'the remainder is one'
    case 2
      m = 'the remainder is two'
    otherwise
      error ('not possible')
    end
```

```
≫ n = 8
the remainder is two
```

3- <u>حلقات التكرار (loops statements)</u>

وهي مجموعة من الجمل تستخدم لتكرار تنفيذ مجموعة من الاوامر لعدد محدد من المرات ، يحتوي برنامج MATLAB على نوعين من حلقات التكرار هما: -

1- حلقات for (for … loops)

2- حلقات while (while … condition)

1- حلقات for :-

وتستخدم لتنفيذ مجموعة من الاوامر مرات متعددة ، ويتم التحكم بعدد مرات تنفيذ الاوامر باستخدام عداد تحدد له قيمتي البداية والنهاية وكذلك مقدار الزيادة، والصيغة العامة لها: -

```
for  i = n : k : m
statement(s)
end
```

حيث ان: i :- قيمة العداد. n :- قيمة البداية للعداد. k :- مقدار الزيادة للعداد. m :- قيمة نهاية العداد.

المثال التالي يجمع الاعداد من 1 الى 10

Ex:-

```
s=0 ; for  i  =  1  :  10
s=s+i  ;  end    disp
(s)
```

- ملاحظة: - تهمل مقدار الزيادة k اذا كانت مقدار زيادة العداد هي 1 ، لان القيمة الافتراضية لبرنامج MATLAB هي 1 .

مثال1 / اكتب برنامج بلغة MATLAB لحساب المعادلات التالية: -

1) $S = \sum_{n=1}^{5000} n$

```
s= 0 ;
for  n = 1 : 5000
s = s + n ;
end
disp ( s)
```

```
>>
 12502500
```

2) $S = \sum_{n=1}^{5000} n^2$

```
s = 0 ;
for  n  =  1  :  5000
s  =  s  +  n  ^  2  ;
end
disp ( s )
```

```
>>
41.679  5000
```

3) $S = \sum_{n=1}^{5000} \frac{1}{n^2}$

```
s = 0 ;
for n = 1 : 5000
s = s + 1 / n^2 ;
end
disp ( s )
```

```
>> 1.6447
```

مثال2 / اكتب برنامج بلغة MATLAB لإيجاد مفكوك العدد !n .

```
% Program Calculate Factorial Number
n = input ( ' the number = ' ) ;
fact = 1 ;
for i = 1 : n
fact = fact * i ;
disp ( [ i   fact  ])
end
```

```
>>          the number = 8
1     1
2     2
3     6
4     24
5     120
6     720
7     5040
8     40320
```

3- حلقات while (while … condition)

تستخدم لتنفيذ امر او مجموعة الاوامر مرات متعددة مادام شرط ما متحقق والصيغة العامة لها: -

```
while  condition
statement(s)
end
```

مثال / البرنامج التالي يجمع الاعداد من 1 الى 10 وبزيادة مقدارها 0.5 مجموع الاعداد

1, 1.5, 2,2.5,3,3.5,…,10

باستخدام while .

```
s = 0 ; i = 1 ;
while  i  <=  10
s = s + i ;
i = i +  0.5 ;
end
display ( 's=',s )
```

```
 >> s= 104.500
```

<u>تمارين</u>

س1 اكتب برنامج بلغة MATLAB لحساب المتسلسلات التالية: -

1) $1^2 + 2^2 + 3^2 + \ldots + 1000^2$

2) $1 - 1/3 + 1/5 - 1/7 + 1/9 - \ldots.$

3) $1/ (1^2 . 3^2) + 1/ (3^2 . 5^2) + 1/ (5^2 . 7^2) + \ldots$

س2 اكتب برنامج بلغة MATLAB لحساب قيمة x من المعادلة التربيعية الاتية:

$$a\,x^2 + b\,x + c = 0$$

$$when\ a = 0\ , x = -\frac{c}{b}$$

Use the quadratic formula $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ when $a \neq 0$

4- **جمل القفز Jamping statements الخروج من جمل التكرار Breaking out of loops**

ان تكرار while يستمر في العمل حتى يصبح الشرط صحيحا او خاطئا ، وان تكرار for يستمر في العمل حتى ينتهي عدد المرات التي تحددها له ، ولكن قد تحتاج في بعض الاحيان الى الخروج المبكر من التكرار قبل انتهائه ، وفي هذه الحالة يمكننا استخدام الامر **break** او **return** ، ويستخدم بالشكل التالي:

```
for / while statement(s)
if condition break
end
…
end
```

مثال / اطبع مربع الاعداد من 1 – 10 لغاية القيمة 36

Sol:

```
clc ; clear ; close all ;
for m = 1 : 10
if ( m ^2 ) >= 36 break ;
end
display ( m ) ;
end
```

```
>> m = 1
m = 2
m = 3
m=4
m=5
```

H.W.

اكتبي البرنامج أعلاه باستخدام while

2) **جملة continue** :

يستخدم الامر continue في داخل الحلقة التكرارية for loop او الحلقة التكرارية المشروطة while loop لانهاء التكرار الحالي، وتجاوز تنفيذ باقي الأوامر ويستخدم بالشكل التالي:

```
for / while statement(s)
if condition continue
end
end
```

مثال/ اطبع القيم الزوجية من 1 – 10

```
clc ; clear ; close all ;
for n = 1 : 10
if rem ( n , 2 ) ~= 0 continue ;
end
display ( n ) ;
end
```