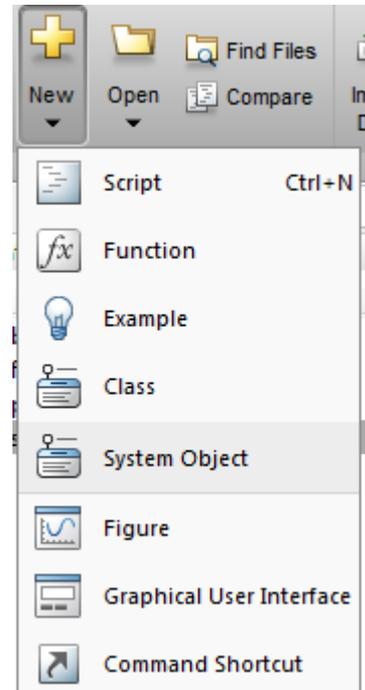


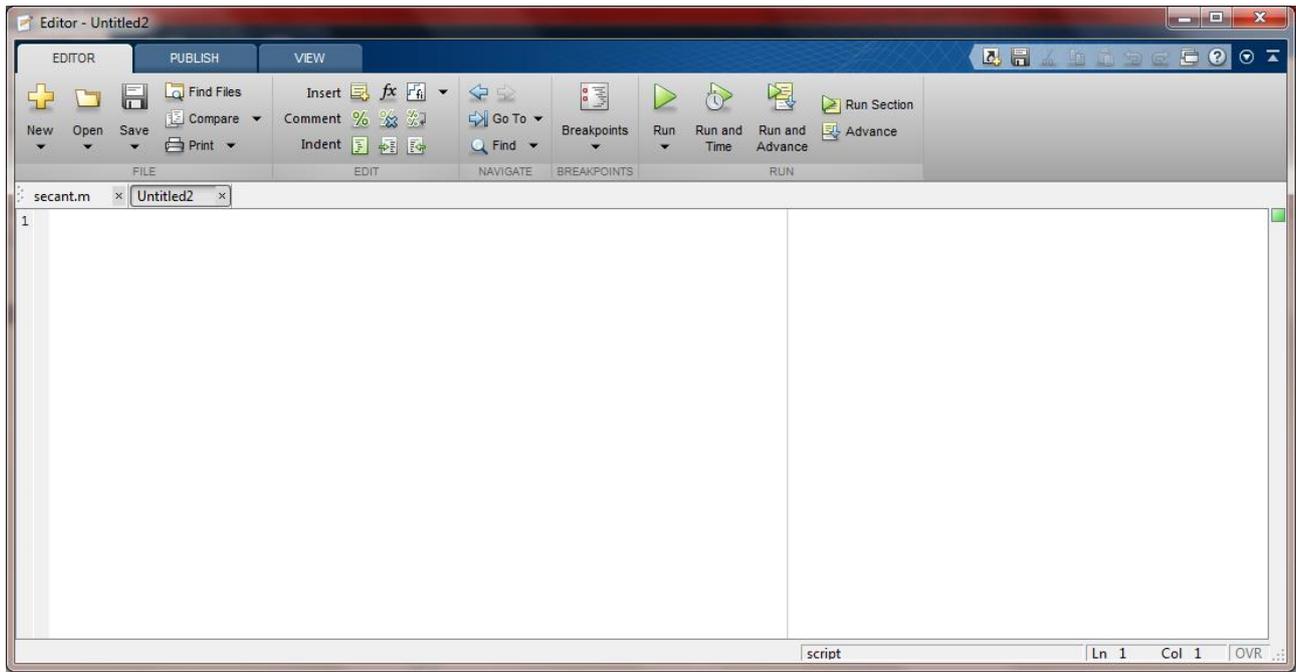
بناء البرامج في MATLAB

إن فائدة البرامج في نظام الـ MATLAB هو لتدوين سلسلة من الأوامر وتنفيذها كلها ككتلة واحدة وفائدة كتابة البرامج هي:

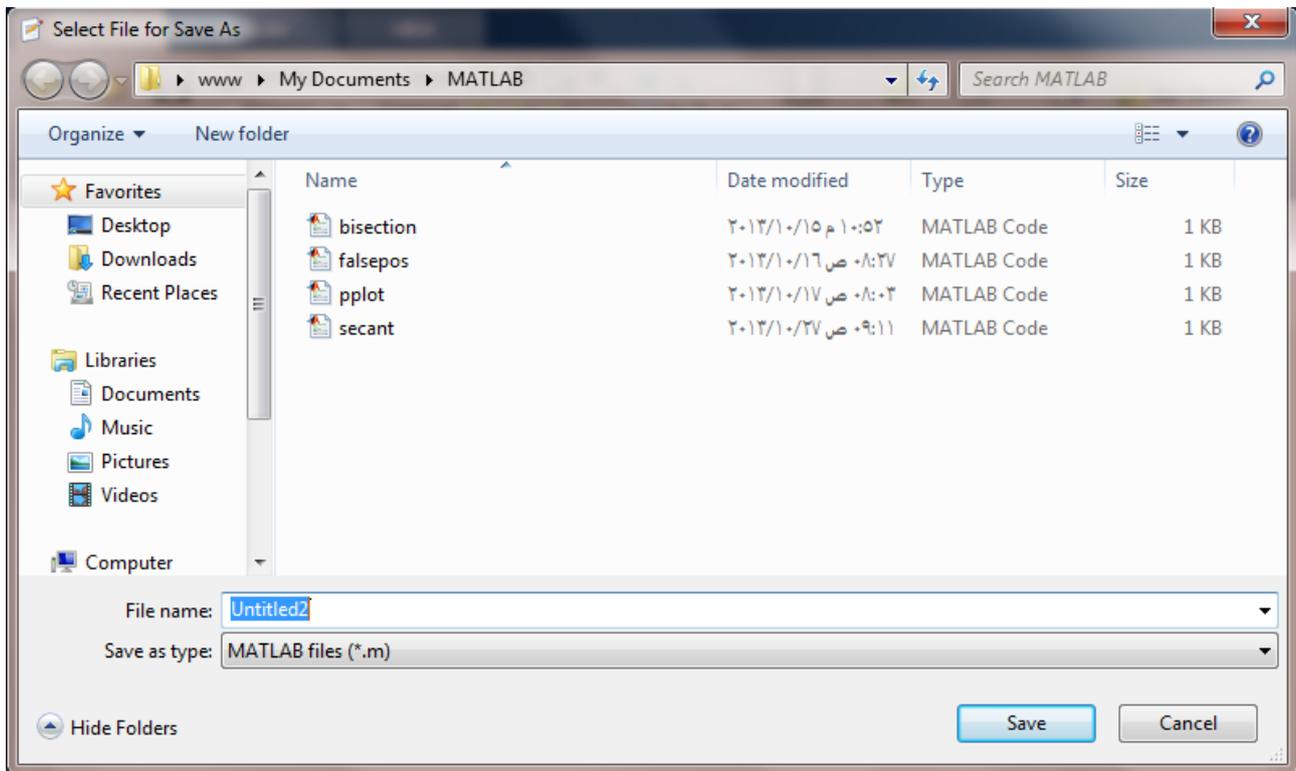
1. إذا أردنا تعديل عنصر، فلا يجب إعادة إدخال الأمر من جديد.
 2. إذا وجد خطأ، فلا يجب إدخال الأمر من جديد.
 3. إذا كتبنا برنامج كبير وأردنا إعادة العملية مرة أخرى فلا يجب إدخال الأوامر من جديد متلافين الأخطاء التي قد تحدث في الإدخال أو في تسلسل إدخال الأوامر.
 4. سهولة تتبع وتصحيح الأخطاء الموجودة Debugging.
- لكتابة برنامج يجب تكوين M-File جديد وذلك من خلال شريط الأدوات الموجود في أعلى الشاشة في نظام الـ MATLAB وكما موضح في الصور أدناه:



لفتح ملف برنامج جديد فارغ نختار NEW من شريط الأدوات ومن القائمة التي تظهر نختار Script أو نختار الأيقونة New Script مباشرة أو من لوحة المفاتيح نضغط المفاتيح Ctrl+N. في كل هذه الحالات سوف تفتح نافذة فارغة نستطيع كتابة ما نريد من الأوامر بها. كما موضح في الصورة التالية:



بعد كتابة سلسلة الأوامر التي نريد تنفيذها في هذه الشاشة الجديدة، نقوم بخزن البرنامج عن طريق الأيقونة Save فيظهر لنا مربع الحوار التالي ومن خلاله نعطي اسماً معيناً للبرنامج ثم نضغط Save.



لتنفيذ البرنامج نضغط على الأيقونة Run، نلاحظ أن شاشة البرنامج سوف تختفي وتظهر نافذة الأوامر ومن خلالها يتم إدخال المدخلات وتظهر فيها النتائج.
لكتابة برنامج بسيط نكتب سلسلة الأوامر التالية في نافذة البرنامج:

```
a=10;
```

```
b=20;
```

```
c=a+b
```

بعد ذلك نحفظ البرنامج من خلال الأيقونة Save ثم ننفذ البرنامج من خلال الخيار Run، نلاحظ أن الناتج وهو $c=30$ سوف يظهر في نافذة الأوامر.

جملة الإدخال input

يمكن إدخال قيم للمتغيرات بعدة طرق إما عن طريق إعطاء قيمة مباشرة مثل $a=28$ أو عن طريق معادلة مثل $y=x*\cos(x)-10$ أو يمكن إدخال القيم عن طريق لوحة المفاتيح أثناء تنفيذ البرنامج.

جملة input هي واحدة من جمل الإدخال التي تقوم بانتظار المستخدم ليدخل قيمة لمتغير معين وصيغتها هي:

```
(عبارة توضيحية ' )=input(اسم المتغير)
```

مثال: `a=input('a=')`

ملاحظة: العبارة التوضيحية تطبع كما هي وفائدتها لتوضيح اسم القيمة المدخلة أثناء التنفيذ.

جملة الإخراج disp

هذه الجملة تفيدنا لعرض قيم متغيرات معينة أو عرض عبارات معينة. وصيغتها كالتالي:

```
disp(اسم المتغير)
```

أو

```
disp('عبارة')
```

في الصيغة الأولى يتم عرض قيمة المتغير المحدد بين القوسين، أما الصيغة الثانية فيتم فيها طباعة العبارة الموجودة بين علامتي الاقتباس كما هي.

مثال: (برنامج نستخدم فيه جملة input وجملة disp)

```
a=input('a= ');
```

```
b=input('b= ');
```

```
c=a+b;
```

```
disp(c)
```

يظهر

سوف

التنفيذ

عند

a=

ينتظر من المستخدم إدخال قيمة a لينتقل البرنامج لتنفيذ الخطوة التالية:

```
a=10
```

```
b=
```

ينتظر الآن إدخال قيمة b:

```
a=10
```

```
b=20
```

```
c=
```

```
30
```

بعد إدخال كلا القيمتين a, b يتم استخراج قيمة c وتقوم جملة disp بعرض الناتج.

جملة inline:

تستخدم هذه الجملة لبناء الدوال وصيغتها هي:

```
inline('صيغة')=اسم المتغير
```

```
f=inline('x^4-4*x^2+6')
```

مثال:

المدخل لهذه الدالة هو المتغير x فلذلك عند استخدام هذه الدالة نستخدم متغير مثلاً

$$y=f(x)$$

أو $a=f(b)$

جملة if:

وهذه الجملة شرطية صيغتها العامة هي:

اعتماداً على صحة الشرط بعد if تنفذ الجمل من 1 إلى n أو لا تنفذ { شرط if
الموجود }

جملة 1

جملة 2

⋮

جملة n

إذا كان الشرط ما بعد if صادقاً تنفذ الجمل ما بعد if وإلا تنفذ الجمل ما بعد جملة else { else
}

جملة 1 وهي الجمل من 1 إلى m

جملة 2

⋮

جملة m

يجب وجود جملة end في نهاية جملة if لفصلها عن باقي جمل البرنامج

جملة else هي جملة تابعة لجملة ولا تستخدم لوحدها وهي اختيارية optional أي ليس من
الضروري أن تستخدم مع جملة if ولكن تستخدم حسب الحالة التي نقوم ببرمجتها. كلمة end
يجب استخدامها عند الانتهاء من جملة if لكي تفصل الجمل التابعة لجملة if عن باقي جمل
البرنامج.

مثال: لاحظ البرنامج التالي الذي يقارن بين عددين ويجد العدد ذو القيمة الأكبر.

```
a=input('a=');
```

```
b=input('b=');
```

```
if a>b
```

```
    disp('a is max')
```

```
else
```

```
    disp('b is max');
```

end

جملة for:

وهي جملة تكرارية تستخدم لتكرار مجموعة من الجمل n من المرات اعتماداً على قيمة العداد. الصيغة العامة لهذه الجملة هي:

for القيمة النهائية: مقدار الزيادة: قيمة أولية= عداد

جملة 1

جملة 2

⋮

جملة n

end

إن قيمة العداد تبدأ بقيمة تسمى القيمة الابتدائية **initial value** وتنتهي بقيمة تسمى القيمة النهائية **final value** وتنتقل قيمة العداد من القيمة الابتدائية إلى القيمة النهائية عن طريق زيادة قيمة العداد في نهاية كل تكرار بقيمة تسمى مقدار الزيادة أو الخطوة **step**. إذا لم يتم تحديد مقدار الزيادة فإن مقدار الزيادة يكون بمقدار 1 في كل تكرار. هذا يعني أن عدد التكرارات يعتمد على قيمة العداد.

مثال: اكتب برنامج يقوم بطباعة الأرقام من 1 إلى n.

```
n=input('n=');
```

```
for i=1:n
```

```
    disp (i)
```

```
end
```