MATLAB INTRODUCTION:

Windows and there purpose:

1. Command window:

It is the main window here we enter variables and then runs programs.

2. Editor window:

Create and debugs script or function files.

3.Figure window:

This window is used for Graphical explanation.

4.Help window:

This window provides help information.

5.Command History window:

Logs commands enter in command window.

6.Workspace window:

It provides information about variables that are used.

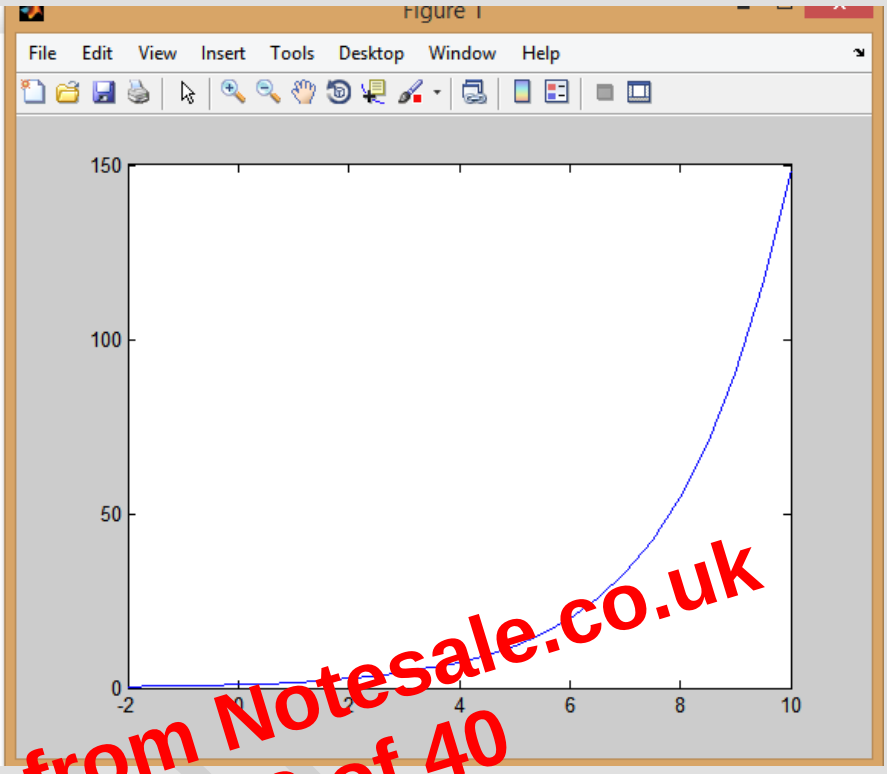
7.Current folder window:

It shows the files in the current folder.

Preview from Notesale.co.uk
Page 2 of 40
DO NOT COPY

Q#4 Y=exp(x/2)

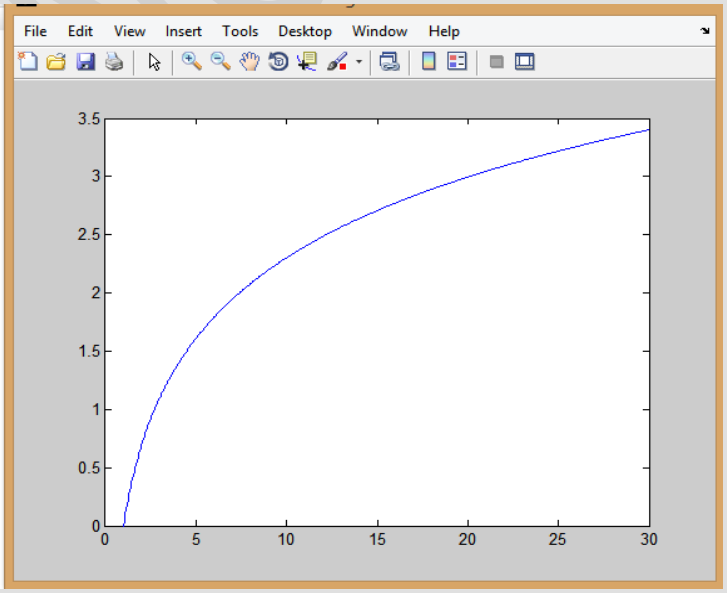
```
Command Window
>> clear all;
>> %plot y=e^x/2
>> X = -2:0.5:10;
   Y = exp(X/2);
   plot(X,Y)
fx >>
```



Preview from Notesale.co.uk
Page 13 of 40

Q#5 Y=log(x)

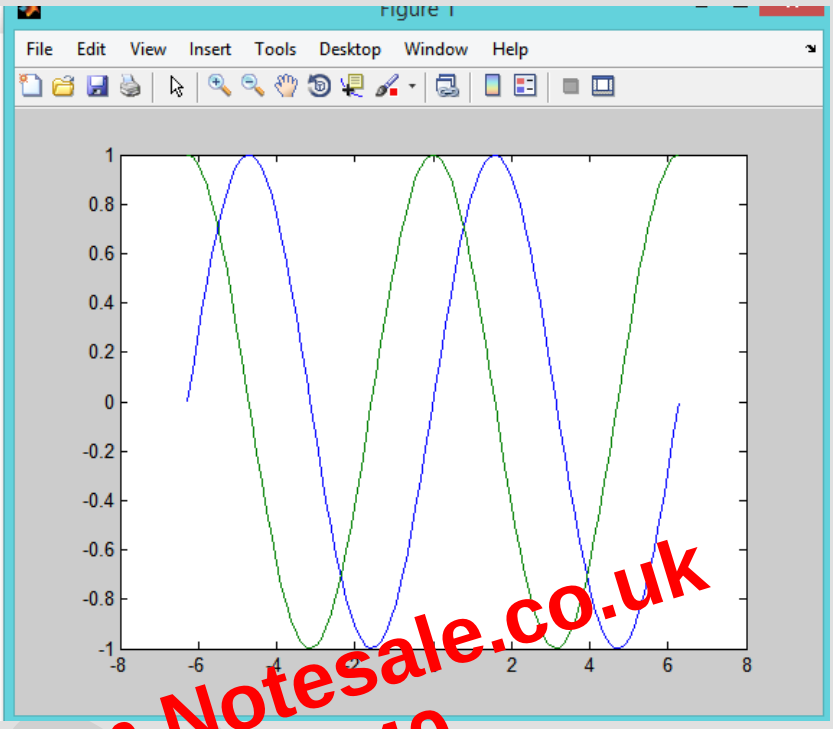
```
Command Window
>> clear all;
>> x=[1:0.05:30];
>> y=log(x);
>> plot(x,y)
fx >>
```



Q#9 Y1=cos(x),y2=sin(x)

Command Window

```
>> clear all;
>> x = linspace(-2*pi,2*pi);
y1 = sin(x);
y2 = cos(x);
p = plot(x,y1,x,y2);
>>
```



Preview from Notesale.co.uk
Page 16 of 40

Q#3 F=x+2cosx

```
% Bisection Algorithm
f = @(x)+(2*cos(x));
a = input('Please enter lower limit, a:  ');
b = input('Please enter upper limit, b:  ');
n = input('Please enter no. of iterations, n: ');
tol = input('Please enter tolerance, tol:  ');
fa = f(a); fb = f(b);
i = 1;
while i <= n
  c = (b - a) / 2.0;
  p = a + c;
  fp = f(p);
  if abs(fp) < 1.0e-20 | c < tol
      fprintf('\nApproximate solution p = %11.8f \n \n',p);
      break;
  else
      i = i+1;
      if fa*fp > 0
         a = p;
         fa = fp;
      else
         b = p;
         fb = fp;
      end
  end
end
```

Preview from Notesale.co.uk
Page 19 of 40

```
Please enter lower limit, a:  1
Please enter upper limit, b:  2
Please enter no. of iterations, n: 15
Please enter tolerance, tol:  0.0001

Approximate solution p =  1.57073975

>>
```

NEWTON-RAPHSON METHOD:

Find the root of y=cos(x) from 0 to pi.

```
% Newton-Raphson Algorithm
 % Find the root of y=cos(x) from o to pi.
 f = @(x) (cos(x));
 fd = @(x) (-sin(x));
 p0 = input('Enter initial approximaation: ');
 n = input('Enter no. of iterations, n: ');
 tol = input('Enter tolerance, tol:  ');
 i = 1;
while i <= n
    d=f(p0)/fd(p0);
    p0 = p0 - d;
    if abs(d) < tol
       fprintf('\nApproximate solution xn= %11.8f \n\n',p0);
       break;
    else
       i = i+1;
    end
end
```

Preview from Notesale.co.uk
Page 22 of 40

OUTPUT:

```
Enter initial approximaation: 1
Enter no. of iterations, n: 20
Enter tolerance, tol:  0.0001

Approximate solution xn=  1.57079633

fx >>
```

OUTPUT:

```
A =

     5    -2
     1    -8
     6     4


b =

    -1
     2
     3


x =

     0
     0

Solution of the system is :
-0.315775
-0.289472
fx >>
```

Preview from Notesale.co.uk
Page 29 of 40