```
function x = NewtonsMethod(f,fd,x0,nmax,epsilon,delta)
   x = x0;
   y= f(x);
   for n=1:nmax
       n
       x = x - y/fd(x)
       y = f(x)
       d= y/fd(x);
       if abs(y) < delta
           return;
       end
       if abs(d) < epsilon
           return;
       end
   end
```

1)
Func1 -@(x)x-2*sin(x)

Deriv1- @(x)1-2*cos(x)

>> Newtons_Method(func1,deriv1,2,10,10^-4,10^-6)

n =

1

x =

1.900995594203909

y =

0.009040087140610

n =

2

x =

1.895511645379595

y =

2.846679920542883e-05

ans =

1.895511645379595

2)

func2-@(x)x^3-sin(x)-7

deriv2-@(x)5*x^4+2*x-21*x^2

>> Newtons_Method(func2,deriv2,4.5,10,10^-4,10^-6)

n =

1

x =

3.103979344402526

y =

22.868267388489265

n =

2

x =

2.339240183435312

y =

5.081433611448389

n =

3

x =

2.042273978628609

y =

0.627187719692039

n =

4

x =

1.993905437937368

y =

0.015271134424061

n =

5

x =

1.992667663448553

y =

9.861225946039553e-06

ans =

1.992667663448553

3)

func3-@(x)sin(x)-1+x

deriv3-@(x)cos(x)+1

>> Newtons_Method(func3,deriv3,2.5,10,10^-4,10^-6)

n =

1

x =

-8.052701890238952

y =

-10.033021911293339

n =

2

x =

4.448180995812336

y =

2.482881361467343

n =

3

x =

1.087737133833393

y =

0.973315236673129

n =

4

x =

0.423126993132994

y =

-0.166259329171142

n =

5

x =

0.510091377732562

```
y =

  -0.001651628008540


n =

     6


x =

   0.510973327911837


y =

    -1.899917309033938e-07


ans =

   0.510973327911837
```

4)

```
func4-@(x)x^5+x^2-1-7*x^3

deriv4-@(x)5*x^4+2*x-21*x^2

Newtons_Method(func4,deriv4,4.5,10,10^-4,10^-6)

n =

     1


x =

   3.749321093899407
```

y =

3.850233466147472e+02

n =

2

x =

3.199560071303805

y =

1.152695079849034e+02

n =

3

x =

2.834110060399900

y =

30.528805890202875

n =

4

x =

2.642793260787940

y =

5.695600036370053

n =

5

x =

2.587237414910560

y =

0.390845821168810

n =

6

x =

2.582828020849536

y =

0.002325128224598

ans =

2.582828020849536