In [1]:
```
import matplotlib.pyplot as plt
import numpy as np
```

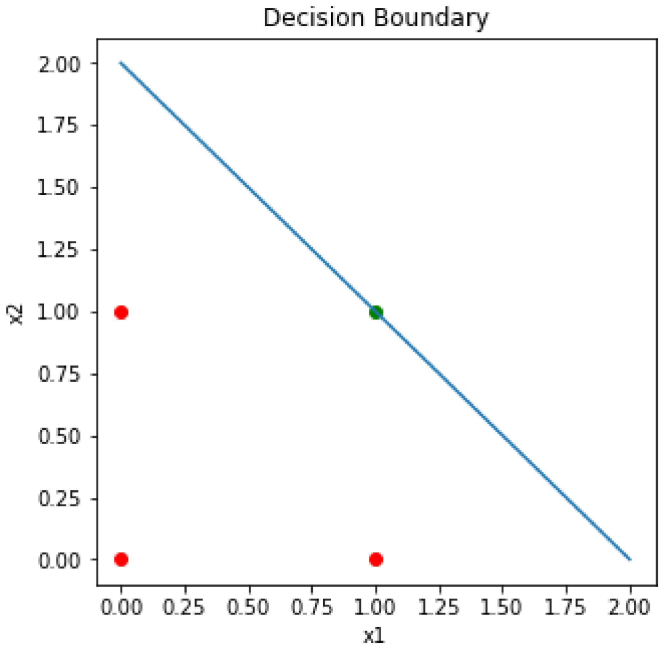
In [2]:
```
#AND
x1=[0,0,1,1]
x2=[0,1,0,1]
def f(error):
    if(len(error)==4):
        if(all(e == 0 for e in error)):
            return 1
        return 0
    return 0
w1,w2,b=-1,0.5,-0.5
print("|     epoch     ino      w1       w2       b     D       AO      E      w1f      w2f      bf    |")
arr=[[0,0],[0,1],[1,0],[1,1]]
d=[0,0,0,1]
s=[0,0,0,0]
t=[0,0,0,0]
error=[]
a=[]
epoch=1
while(f(error)==0):
    for i in range(len(arr)):
        s[i]=b+arr[i][0]*w1+arr[i][1]*w2
        if(s[i]>=0):
            y=1
        else:
            y=0
        t[i]=d[i]-y
        if(t[i]!=0):
            w1n=w1+t[i]*arr[i][0]
            w2n=w2+t[i]*arr[i][1]
            bn=b+t[i]
            w1=w1n
            w2=w2n
            b=bn
            print("|\t",epoch,"\t",i+1,"\t",w1,"\t",w2,"\t",b,"\t",d[i],"\t",y,"\t",t[i],"\t",w1n,"\t",w2n,"\t",bn,"\t|")
        else:
            print("|\t",epoch,"\t",i+1,"\t",w1,"\t",w2,"\t",b,"\t",d[i],"\t",y,"\t",t[i],"\t",w1,"\t",w2,"\t",b,"\t|")
            error.append(0)
    mse=[x*x for x in t]
```

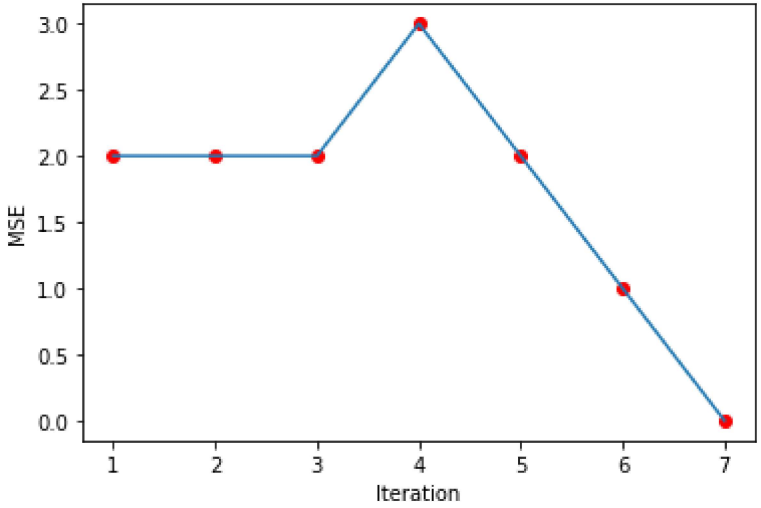
```
    x=sum(mse)
    a.append(x)
    if(f(error)):
        break
    else:
        epoch+=1
        print("-----------------------------------------------------------------------------------------------")
        error.clear()
if(f(error)):
    print("converges")
else:
    print("not converges")
def decisionboundary(l,x1,x2,d,w1,w2):
    plt.figure(figsize=(5,5))
    plt.title("Decision Boundary")
    for i in range(l):
        if d[i]==1:
            color="g"
        else:
            color="r"
        plt.scatter(x1[i],x2[i],c=color)
    x=np.linspace(0,2,4)
    y=-x+2
    plt.plot(x,y)
    plt.xlabel('x1')
    plt.ylabel('x2')
    plt.show()
decisionboundary(len(arr),x1,x2,d,w1,w2)
def mse(a):
    plt.plot(list(range(1,len(a)+1)),a, 'ro')
    plt.plot(list(range(1,len(a)+1)),a)
    plt.ylabel('MSE')
    plt.xlabel('Iteration')
    plt.show()
mse(a)
```

epoch	ino	w1	w2	b	D	AO	E	w1f	w2f	bf
1	1	-1	0.5	-0.5	0	0	0	-1	0.5	-0.5
1	2	-1	-0.5	-1.5	0	1	-1	-1	-0.5	-1.5
1	3	-1	-0.5	-1.5	0	0	0	-1	-0.5	-1.5
1	4	0	0.5	-0.5	1	0	1	0	0.5	-0.5
2	1	0	0.5	-0.5	0	0	0	0	0.5	-0.5
2	2	0	-0.5	-1.5	0	1	-1	0	-0.5	-1.5
2	3	0	-0.5	-1.5	0	0	0	0	-0.5	-1.5

2	4	1	0.5	-0.5	1	0	1	1	0.5	-0.5
3	1	1	0.5	-0.5	0	0	0	1	0.5	-0.5
3	2	1	-0.5	-1.5	0	1	-1	1	-0.5	-1.5
3	3	1	-0.5	-1.5	0	0	0	1	-0.5	-1.5
3	4	2	0.5	-0.5	1	0	1	2	0.5	-0.5
4	1	2	0.5	-0.5	0	0	0	2	0.5	-0.5
4	2	2	-0.5	-1.5	0	1	-1	2	-0.5	-1.5
4	3	1	-0.5	-2.5	0	1	-1	1	-0.5	-2.5
4	4	2	0.5	-1.5	1	0	1	2	0.5	-1.5
5	1	2	0.5	-1.5	0	0	0	2	0.5	-1.5
5	2	2	0.5	-1.5	0	0	0	2	0.5	-1.5
5	3	1	0.5	-2.5	0	1	-1	1	0.5	-2.5
5	4	2	1.5	-1.5	1	0	1	2	1.5	-1.5
6	1	2	1.5	-1.5	0	0	0	2	1.5	-1.5
6	2	2	0.5	-2.5	0	1	-1	2	0.5	-2.5
6	3	2	0.5	-2.5	0	0	0	2	0.5	-2.5
6	4	2	0.5	-2.5	1	1	0	2	0.5	-2.5
7	1	2	0.5	-2.5	0	0	0	2	0.5	-2.5
7	2	2	0.5	-2.5	0	0	0	2	0.5	-2.5
7	3	2	0.5	-2.5	0	0	0	2	0.5	-2.5
7	4	2	0.5	-2.5	1	1	0	2	0.5	-2.5

converges





In [3]:

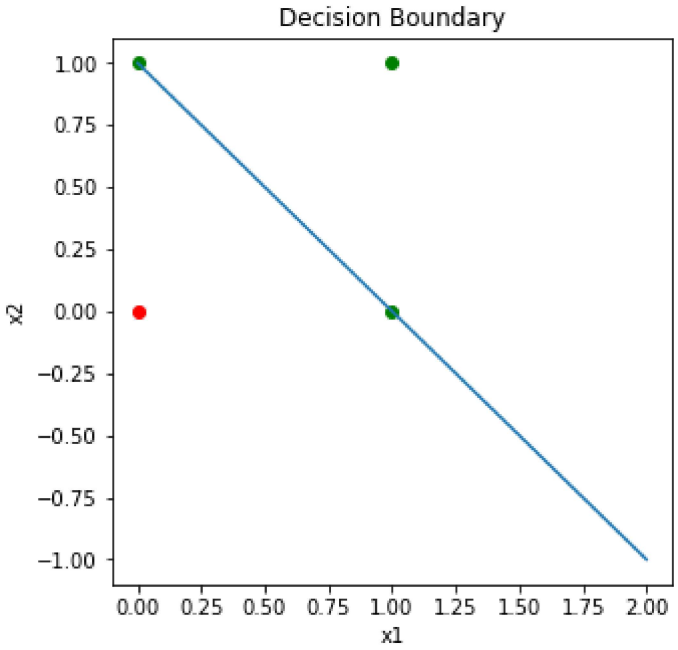
```
#OR
def f(error):
    if(len(error)==4):
        if(all(e == 0 for e in error)):
            return 1
```

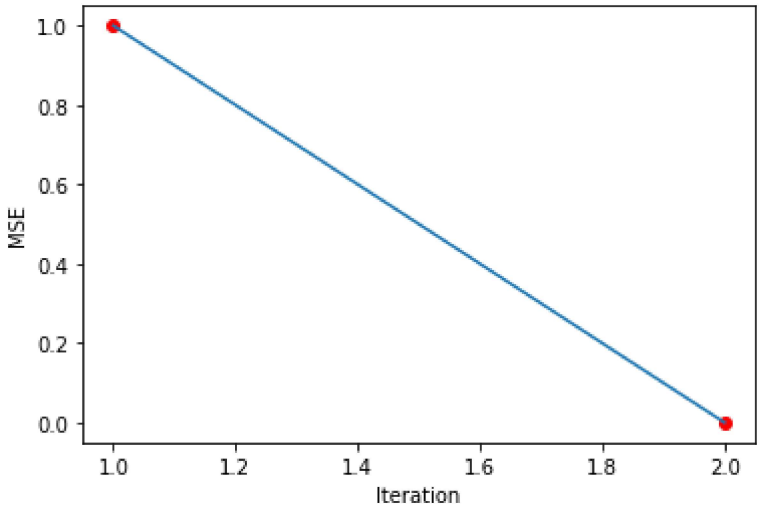
```
        return 0
    return 0
w1,w2,b=1,1,0
print("|     epoch     ino      w1       w2       b     D       AO      E      w1f      w2f      bf    |")
arr=[[0,0],[0,1],[1,0],[1,1]]
d=[0,1,1,1]
s=[0,0,0,0]
t=[0,0,0,0]
error=[]
epoch=1
a=[]
while(f(error)==0):
    for i in range(len(arr)):
        s[i]=b+arr[i][0]*w1+arr[i][1]*w2
        if(s[i]>=0):
            y=1
        else:
            y=0
        t[i]=d[i]-y
        if(t[i]!=0):
            w1n=w1+t[i]*arr[i][0]
            w2n=w2+t[i]*arr[i][1]
            bn=b+t[i]
            w1=w1n
            w2=w2n
            b=bn
            print("|\t",epoch,"\t",i+1,"\t",w1,"\t",w2,"\t",b,"\t",d[i],"\t",y,"\t",t[i],"\t",w1n,"\t",w2n,"\t",bn,"\t|")
        else:
            print("|\t",epoch,"\t",i+1,"\t",w1,"\t",w2,"\t",b,"\t",d[i],"\t",y,"\t",t[i],"\t",w1,"\t",w2,"\t",b,"\t|")
            error.append(0)
    mse=[x*x for x in t]
    x=sum(mse)
    a.append(x)
    if(f(error)):
        break
    else:
        epoch+=1
        print("-------------------------------------------------------------------------------------------")
        error.clear()
if(f(error)):
    print("converges")
else:
    print("not converges")
def decisionboundary(l,x1,x2,d,w1,w2):
```

```
    plt.figure(figsize=(5,5))
    plt.title("Decision Boundary")
    for i in range(l):
        if d[i]==1:
            color="g"
        else:
            color="r"
        plt.scatter(x1[i],x2[i],c=color)
    x=np.linspace(0,2,4)
    y=-x+1
    plt.plot(x,y)
    plt.xlabel('x1')
    plt.ylabel('x2')
    plt.show()
decisionboundary(len(arr),x1,x2,d,w1,w2)
def mse(a):
    plt.plot(list(range(1,len(a)+1)),a, 'ro')
    plt.plot(list(range(1,len(a)+1)),a)
    plt.ylabel('MSE')
    plt.xlabel('Iteration')
    plt.show()
mse(a)
```

epoch	ino	w1	w2	b	D	AO	E	w1f	w2f	bf
1	1	1	1	-1	0	1	-1	1	1	-1
1	2	1	1	-1	1	1	0	1	1	-1
1	3	1	1	-1	1	1	0	1	1	-1
1	4	1	1	-1	1	1	0	1	1	-1
2	1	1	1	-1	0	0	0	1	1	-1
2	2	1	1	-1	1	1	0	1	1	-1
2	3	1	1	-1	1	1	0	1	1	-1
2	4	1	1	-1	1	1	0	1	1	-1

converges





In [4]:

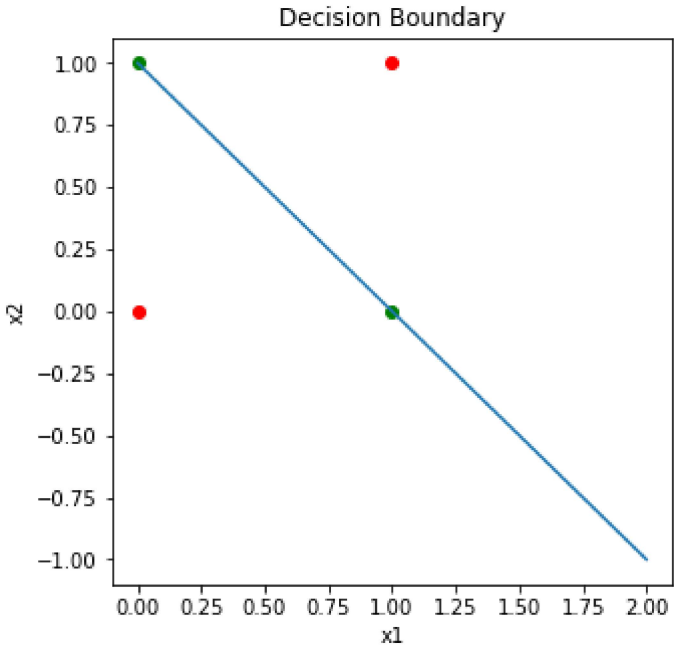
```
#xor
def f(error):
    if(len(error)==4):
        if(all(e == 0 for e in error)):
            return 1
```

```
        return 0
    return 0
w1,w2,b=1,1,0
print("|     epoch     ino      w1       w2       b     D       AO      E      w1f      w2f      bf    |")
arr=[[0,0],[0,1],[1,0],[1,1]]
d=[0,1,1,0]
s=[0,0,0,0]
t=[0,0,0,0]
a=[]
error=[]
epoch=1
while(f(error)==0):
    for i in range(len(arr)):
        s[i]=b+arr[i][0]*w1+arr[i][1]*w2
        if(s[i]>=0):
            y=1
        else:
            y=0
        t[i]=d[i]-y
        if(t[i]!=0):
            w1n=w1+t[i]*arr[i][0]
            w2n=w2+t[i]*arr[i][1]
            bn=b+t[i]
            w1=w1n
            w2=w2n
            b=bn
            print("|\t",epoch,"\t",i+1,"\t",w1,"\t",w2,"\t",b,"\t",d[i],"\t",y,"\t",t[i],"\t",w1n,"\t",w2n,"\t",bn,"\t|")
        else:
            print("|\t",epoch,"\t",i+1,"\t",w1,"\t",w2,"\t",b,"\t",d[i],"\t",y,"\t",t[i],"\t",w1,"\t",w2,"\t",b,"\t|")
            error.append(0)
    if(epoch==5):
        break
    else:
        epoch+=1
        print("-------------------------------------------------------------------------------------------")
        error.clear()
if(f(error)):
    print("converges")
else:
    print("not converges")
c=error.count(0)
a=c*100/epoch
print("Accuracy",a)
def decisionboundary(l,x1,x2,d,w1,w2):
```

```
    plt.figure(figsize=(5,5))
    plt.title("Decision Boundary")
    for i in range(l):
        if d[i]==1:
            color="g"
        else:
            color="r"
        plt.scatter(x1[i],x2[i],c=color)
    x=np.linspace(0,2,4)
    y=-x+1
    plt.plot(x,y)
    plt.xlabel('x1')
    plt.ylabel('x2')
    plt.show()
decisionboundary(len(arr),x1,x2,d,w1,w2)
```

epoch	ino	w1	w2	b	D	AO	E	w1f	w2f	bf
1	1	1	1	-1	0	1	-1	1	1	-1
1	2	1	1	-1	1	1	0	1	1	-1
1	3	1	1	-1	1	1	0	1	1	-1
1	4	0	0	-2	0	1	-1	0	0	-2
2	1	0	0	-2	0	0	0	0	0	-2
2	2	0	1	-1	1	0	1	0	1	-1
2	3	1	1	0	1	0	1	1	1	0
2	4	0	0	-1	0	1	-1	0	0	-1
3	1	0	0	-1	0	0	0	0	0	-1
3	2	0	1	0	1	0	1	0	1	0
3	3	0	1	0	1	1	0	0	1	0
3	4	-1	0	-1	0	1	-1	-1	0	-1
4	1	-1	0	-1	0	0	0	-1	0	-1
4	2	-1	1	0	1	0	1	-1	1	0
4	3	0	1	1	1	0	1	0	1	1
4	4	-1	0	0	0	1	-1	-1	0	0
5	1	-1	0	-1	0	1	-1	-1	0	-1
5	2	-1	1	0	1	0	1	-1	1	0
5	3	0	1	1	1	0	1	0	1	1
5	4	-1	0	0	0	1	-1	-1	0	0

```
not converges
Accuracy 0.0
```


Decision Boundary
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
x2
0.00 0.25 0.50 0.75 1.00 1.25 1.50 1.75 2.00
x1