Roll no:241801222
Name: Ram kumar .S.S

]:

1 #stngLe layer perceptron import numpy as np

```
X = np.array([[0,0],
[0,1],
[1,0],
[1,1]])
Y = np.array([0, 0, 0, 1])
```

2]:
```
w = np.array([0.3, -0.2])
```

bias = -0.4

[3]:
```
eta=0.2
```

[4]:
```
def activation(net):
    return 1 if net >= 0 else 0
```

[5]:
```
error_count = 0
```

[6]:
```
for i in range(len(X)):
    x = X[i]
    y = Y[i]
```

net = np.dot(x, w) + bias
y_pred - activation (net)
error = Y - y pred

if error e:
w = w + eta error * x bias = bias + eta error error count 1 print(f"Update: Input={x}, Target={y} pred={y_pred Error error New weight= w New Bias bias ' else:
print(f"No change: Input x}, Target={y}, pred={y_pred}") Update: Input=[1 1], Target=l, pred=ø, ')
Error=l, New weight=[ø.5 5 0.], New Bias=-0.2 0.

if error count — converged . ")
print("\nFina1 Results: ") for i
in range(1en(X)):
net = np.dot(X[i], w) + bias y_pred =
activation(net) print(f"lnput: {X[i]} -
+ Output: {y_pred

Final Results:

```
[0 0] → Output: 0
[0 1] → Output: 0
[1 0] → Output: 1
[1 1] → Output: 1
```

Input:
Input:
Input:
Input: