**PROGRAM [4]:**

from collections import defaultdict

class Graph:

def \_\_init\_\_(self):

self.graph = defaultdict(list)

def add\_edge(self, u, v):

self.graph[u].append(v)

def dfs\_util(self, v, visited):

visited[v] = True

print(v, end=" ")

for i in self.graph[v]:

if not visited[i]:

self.dfs\_util(i, visited)

def dfs(self):

V = len(self.graph)

visited = [False] \* V

for i in range(V):

if not visited[i]:

self.dfs\_util(i, visited)

g = Graph()

g.add\_edge(0, 1)

g.add\_edge(0, 2)

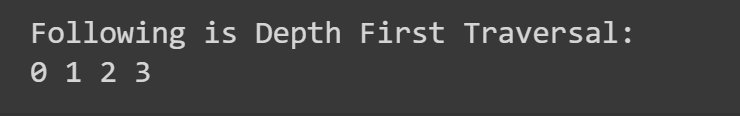
g.add\_edge(1, 2)

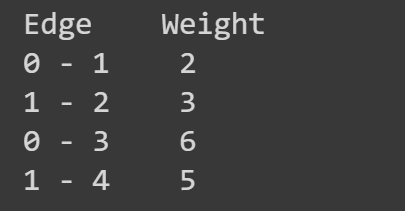
g.add\_edge(2, 0)

g.add\_edge(2, 3)

g.add\_edge(3, 3)

**OUTPUT [4]:**





print("Following is Depth First Traversal:")

g.dfs()