**NAME:GINKA HEMDEEP REGNO:RA1911030010072**

**BFS and DFS**

**Aim:**

To implement Breadth first search and depth first search using python.

**SOURCE CODE:**

**Breadth first search:**

graph = {

'A' : ['B','C'],

'B' : ['D'],

'C' : ['F'],

'D' : ['E', 'F'],

'E' : [],

'F' : ['A']

}

visited = []

queue = []

def bfs(visited, graph, node):

visited.append(node)

queue.append(node)

while queue:

s = queue.pop(0)

print (s, end = " ")

for neighbour in graph[s]:

if neighbour not in visited:

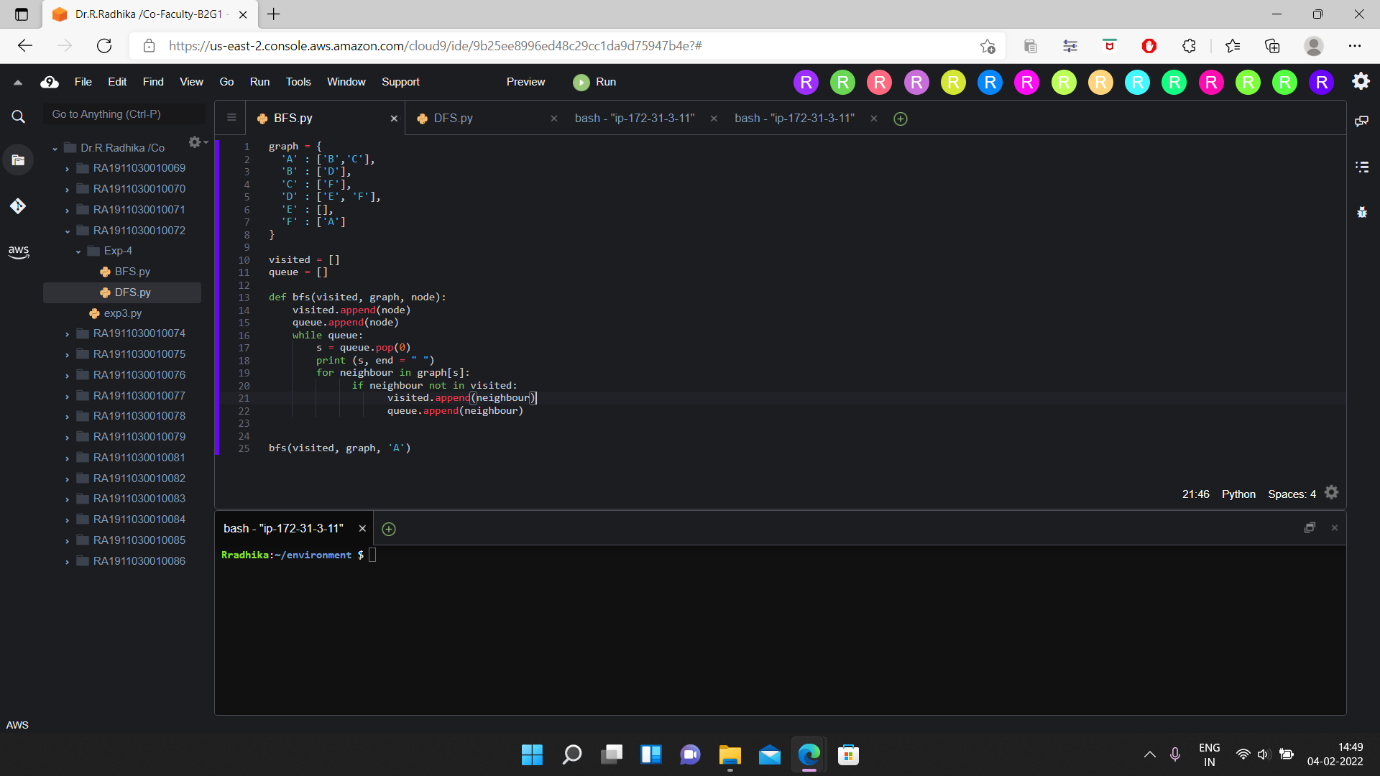
visited.append(neighbour)

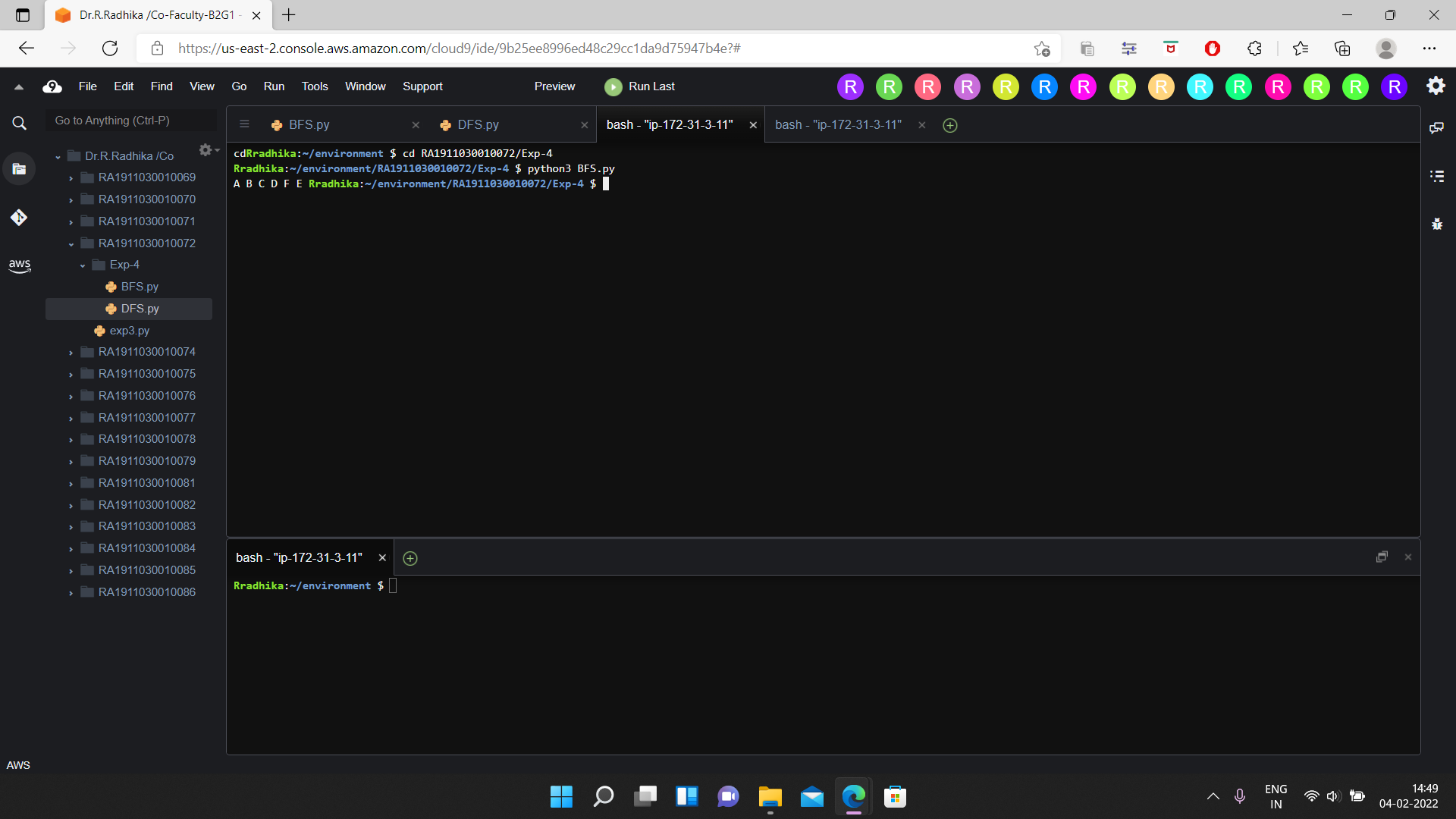
queue.append(neighbour)

bfs(visited, graph, 'A')

**OUTPUT:**

**A B C D F E**

****

****

**SOURCE CODE:**

**DEPTH FIRST SEARCH:**

graph = {

'A' : ['B','C'],

'B' : ['D'],

'C' : ['F'],

'D' : ['E', 'F'],

'E' : [],

'F' : ['A']

}

visited = set()

def dfs(visited, graph, node):

if node not in visited:

print (node)

visited.add(node)

for neighbour in graph[node]:

dfs(visited, graph, neighbour)

dfs(visited, graph, 'A')

**OUTPUT:**

**A**

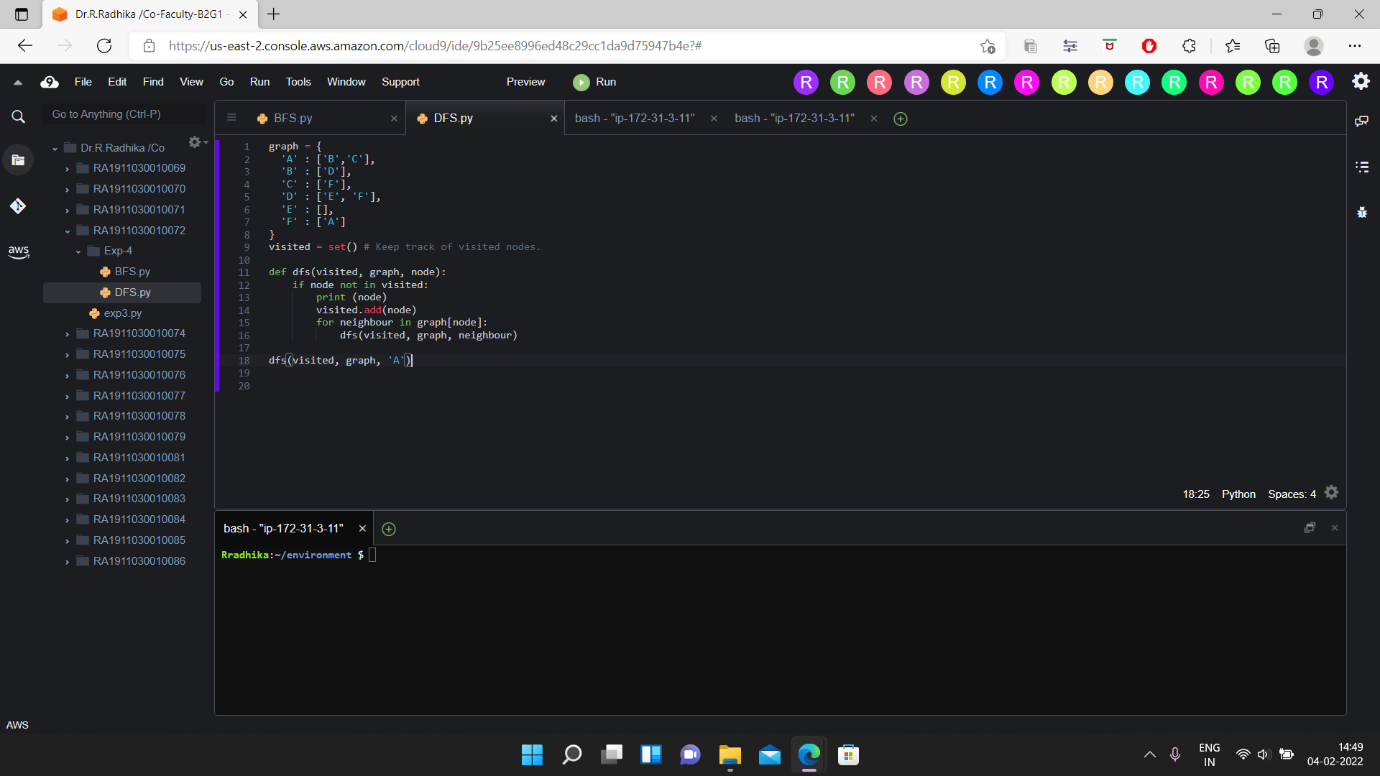
**B**

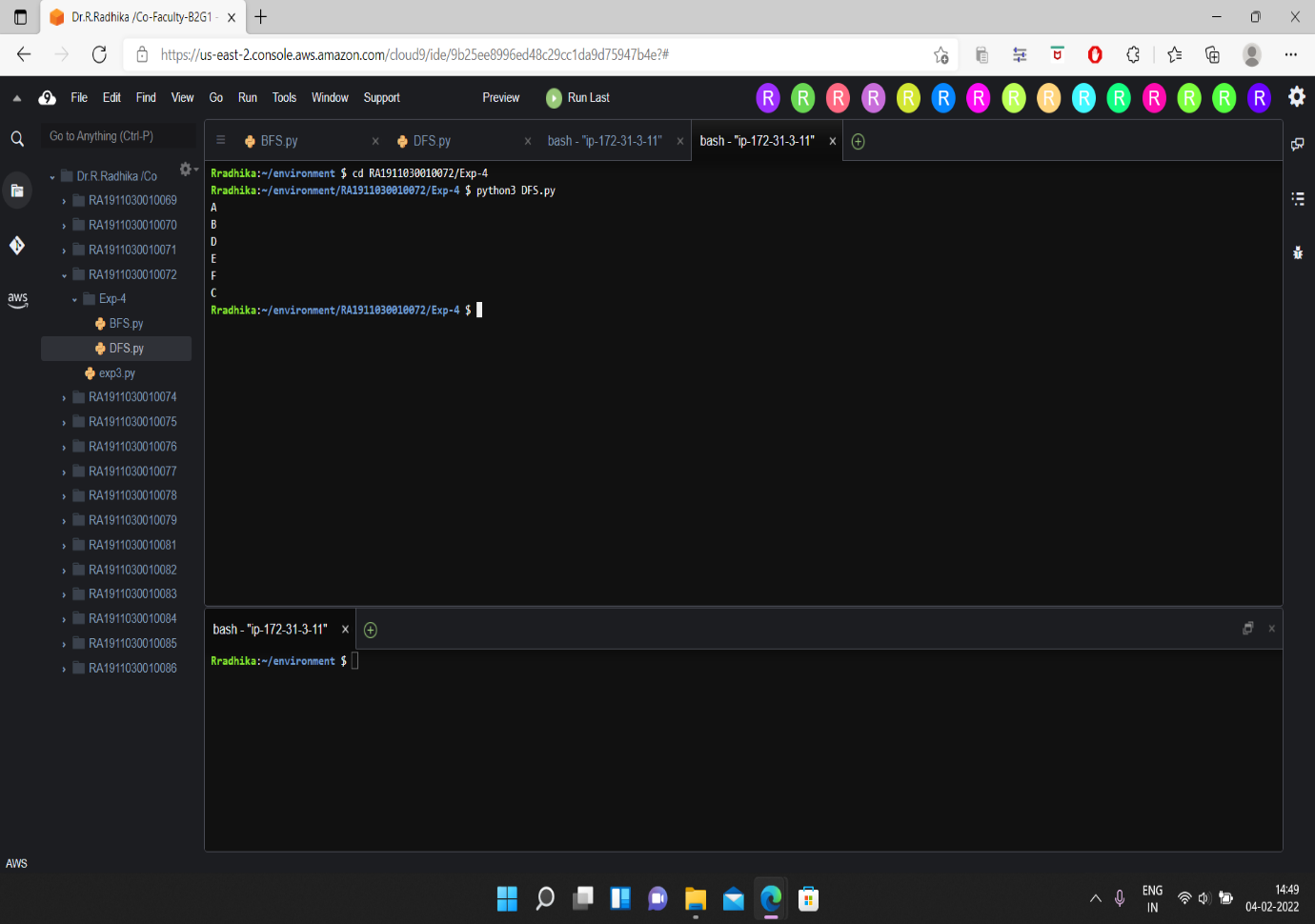
**D**

**E**

**F**

**C**

****

****