***PYTHON CODE FILE-2***

**Q: Develop a python code file for map coloring problem.**

**Ans:**

def is\_safe(graph, colors, node, color):

for neighbor in graph[node]:

if colors[neighbor] == color:

return False

return True

def solve\_map\_coloring(graph, colors, num\_colors, node):

if node == len(graph):

return True

for color in range(1, num\_colors + 1):

if is\_safe(graph, colors, node, color):

colors[node] = color

if solve\_map\_coloring(graph, colors, num\_colors, node + 1):

return True

# Backtrack if assigning color doesn't lead to a solution

colors[node] = 0

return False

def map\_coloring(graph, num\_colors):

# Initialize all nodes with color 0 (uncolored)

colors = [0] \* len(graph)

if not solve\_map\_coloring(graph, colors, num\_colors, 0):

return "Solution does not exist"

return colors

# Example

if \_\_name\_\_ == "\_\_main\_\_":

graph = [

[1, 2], # Neighbors of node 0

[0, 2], # Neighbors of node 1

[0, 1, 3], # Neighbors of node 2

[2] # Neighbors of node 3

]

num\_colors = 3

solution = map\_coloring(graph, num\_colors)

print("Solution:", solution)

**- Varshith**

**- 2320030196**