

# Readme

I have implemented the features of the project by making a class for graph named myGraphClass.

This is the file structure of the project :-

src		
— Main		
— java		
— — — Main.java	-The file runs the command-line interface and interact with myGraphClass.	
— — — myGraphClass.java	-This class generates and interact with the graph via Graphviz, this implements the methods for all the 4 features.	
— — — testGraph.dot	-This file is used in testing for importing the graph and do tests of all the 4 features.	
test		
— — java		
— — — — myGraphClass.java	-This file contains the tests for all 4 features, each test-case tests one feature, I have used assert statements to compare the outputs.	
pom.xml	-This file contains the dependencies of the project.	

Run the **src/Main/java/Main.java** file to access the command-line interface.

To run the test cases run **test/java/myGraphClass.java**

Following are the commands I have used to run my code :-

“src/main/java/testGraph.dot” -> Parses the .dot file

“0” -> To print the graph details

“1” -> To enter the details of the graph to a file

“src/main/java/testGraph.txt” -> File-path for the output

“2” -> To add node to the graph

“node0” -> Label of the node to add

“3” -> To add multiple nodes to the graph

“node1,node2,node3” -> Three nodes to be added to the graph

“4” -> To add an edge to the graph

“white” -> First node of the edge

“node3” -> Second node of the edge

“5” -> To remove a node from the graph

“yellow” -> Label of the node to remove

“6” -> To remove multiple nodes from the graph

“node1,black” -> 2 nodes to remove

“7” -> To remove an edge from the graph

“blue” -> Label of the first node of the edge to remove

“cyan” -> Label of the second node of the edge to remove

“8” -> To output the graph to .dot file

“src/main/java/testGraph\_o.dot” -> Path of the output file

“9” -> To output the graph to an image

“src/main/java/testGraph.png” -> Path of the output image file

“Q” -> To exit the program