

Ethic

src

main

java

Ethic

Project

~\Documents\Ethic

.idea

Development\_Process

README.md

written\_work.jpg

written\_work.pdf

InputFiles

sample.txt

sample\_II.txt

src

main

java

100% classes, 92

Ethic

88% methods

Ethic.class

Graph

100% metho

GraphNode.class

Graph.class

sample.txt

resources

test

java

Unit\_Test\_I

target

Ethic.iml

pom.xml

README.md

External Libraries

< 1.8 >

Library\Java\JavaVi

ant-javafx.jar

library root

charsets.jar

library root

clrddata.jar

library root

deploy.jar

library root

dnsns.jar

library root

dt.jar

library root

h.java

Ethic.java

tokenString

Q

tokenString

13 results

Aa

W

\*

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

```
import java.util.*;
import java.io.*;
import java.io.IOException;
import java.nio.file.Path;
import java.nio.file.Paths;

public class Ethic {
    public Graph graph;

    public Ethic() {
        System.out.println("Graph Has Been Initialized");
        graph = new Graph();
    }

    public void createGraph(List<String> routes) {
        // Trim all leading & trailing white spaces
        List<String> trimRoutes = new ArrayList<String>();
        for (String route: routes) {
            trimRoutes.add(route.trim());
        }
        routes = trimRoutes;

        // Create Nodes
        for (char id: getDistinctNodes(routes)) {
            graph.addNode(id);
        }

        // Create Edges
        for (String route: routes) {
```

Unit\_Test\_I.java

sample.txt

Development\_Process\README.md

Ethic.java

tokenString

Q

tokenString

13 results

Aa

W

\*

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

```
import java.util.*;
import java.io.*;
import java.io.IOException;
import java.nio.file.Path;
import java.nio.file.Paths;

public class Ethic {
    public Graph graph;

    public Ethic() {
        System.out.println("Graph Has Been Initialized");
        graph = new Graph();
    }

    public void createGraph(List<String> routes) {
        // Trim all leading & trailing white spaces
        List<String> trimRoutes = new ArrayList<String>();
        for (String route: routes) {
            trimRoutes.add(route.trim());
        }
        routes = trimRoutes;

        // Create Nodes
        for (char id: getDistinctNodes(routes)) {
            graph.addNode(id);
        }

        // Create Edges
        for (String route: routes) {
```

Unit\_Test\_I.java

sample.txt

Development\_Process\README.md

Ethic.java

tokenString

Q

tokenString

13 results

Aa

W

\*

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

```
import java.util.*;
import java.io.*;
import java.io.IOException;
import java.nio.file.Path;
import java.nio.file.Paths;

public class Ethic {
    public Graph graph;

    public Ethic() {
        System.out.println("Graph Has Been Initialized");
        graph = new Graph();
    }

    public void createGraph(List<String> routes) {
        // Trim all leading & trailing white spaces
        List<String> trimRoutes = new ArrayList<String>();
        for (String route: routes) {
            trimRoutes.add(route.trim());
        }
        routes = trimRoutes;

        // Create Nodes
        for (char id: getDistinctNodes(routes)) {
            graph.addNode(id);
        }

        // Create Edges
        for (String route: routes) {
```

Unit\_Test\_I.java

sample.txt

Development\_Process\README.md

Ethic.java

tokenString

Q

tokenString

13 results

Aa

W

\*

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

```
import java.util.*;
import java.io.*;
import java.io.IOException;
import java.nio.file.Path;
import java.nio.file.Paths;

public class Ethic {
    public Graph graph;

    public Ethic() {
        System.out.println("Graph Has Been Initialized");
        graph = new Graph();
    }

    public void createGraph(List<String> routes) {
        // Trim all leading & trailing white spaces
        List<String> trimRoutes = new ArrayList<String>();
        for (String route: routes) {
            trimRoutes.add(route.trim());
        }
        routes = trimRoutes;

        // Create Nodes
        for (char id: getDistinctNodes(routes)) {
            graph.addNode(id);
        }

        // Create Edges
        for (String route: routes) {
```

Unit\_Test\_I.java

sample.txt

Development\_Process\README.md

Ethic.java

tokenString

Q

tokenString

13 results

Aa

W

\*

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

```
import java.util.*;
import java.io.*;
import java.io.IOException;
import java.nio.file.Path;
import java.nio.file.Paths;

public class Ethic {
    public Graph graph;

    public Ethic() {
        System.out.println("Graph Has Been Initialized");
        graph = new Graph();
    }

    public void createGraph(List<String> routes) {
        // Trim all leading & trailing white spaces
        List<String> trimRoutes = new ArrayList<String>();
        for (String route: routes) {
            trimRoutes.add(route.trim());
        }
        routes = trimRoutes;

        // Create Nodes
        for (char id: getDistinctNodes(routes)) {
            graph.addNode(id);
        }

        // Create Edges
        for (String route: routes) {
```

Unit\_Test\_I.java

sample.txt

Development\_Process\README.md

Ethic.java

tokenString

Q

tokenString

13 results

Aa

W

\*

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

```
import java.util.*;
import java.io.*;
import java.io.IOException;
import java.nio.file.Path;
import java.nio.file.Paths;

public class Ethic {
    public Graph graph;

    public Ethic() {
        System.out.println("Graph Has Been Initialized");
        graph = new Graph();
    }

    public void createGraph(List<String> routes) {
        // Trim all leading & trailing white spaces
        List<String> trimRoutes = new ArrayList<String>();
        for (String route: routes) {
            trimRoutes.add(route.trim());
        }
        routes = trimRoutes;

        // Create Nodes
        for (char id: getDistinctNodes(routes)) {
            graph.addNode(id);
        }

        // Create Edges
        for (String route: routes) {
```

Unit\_Test\_I.java

sample.txt

Development\_Process\README.md

Ethic.java

tokenString

Q

tokenString

13 results

Aa

W

\*

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

```
import java.util.*;
import java.io.*;
import java.io.IOException;
import java.nio.file.Path;
import java.nio.file.Paths;

public class Ethic {
    public Graph graph;

    public Ethic() {
        System.out.println("Graph Has Been Initialized");
        graph = new Graph();
    }

    public void createGraph(List<String> routes) {
        // Trim all leading & trailing white spaces
        List<String> trimRoutes = new ArrayList<String>();
        for (String route: routes) {
            trimRoutes.add(route.trim());
        }
        routes = trimRoutes;

        // Create Nodes
        for (char id: getDistinctNodes(routes)) {
            graph.addNode(id);
        }

        // Create Edges
        for (String route: routes) {
```

Unit\_Test\_I.java

sample.txt

Development\_Process\README.md

Ethic.java

tokenString

Q

tokenString

13 results

Aa

W

\*

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26