
Data Structures and Algorithms

COMP 103

2019-20

Semester 2

Lecture 06b

Dr. Kerese Manueli

kerese.manueli@ecs.vuw.ac.nz

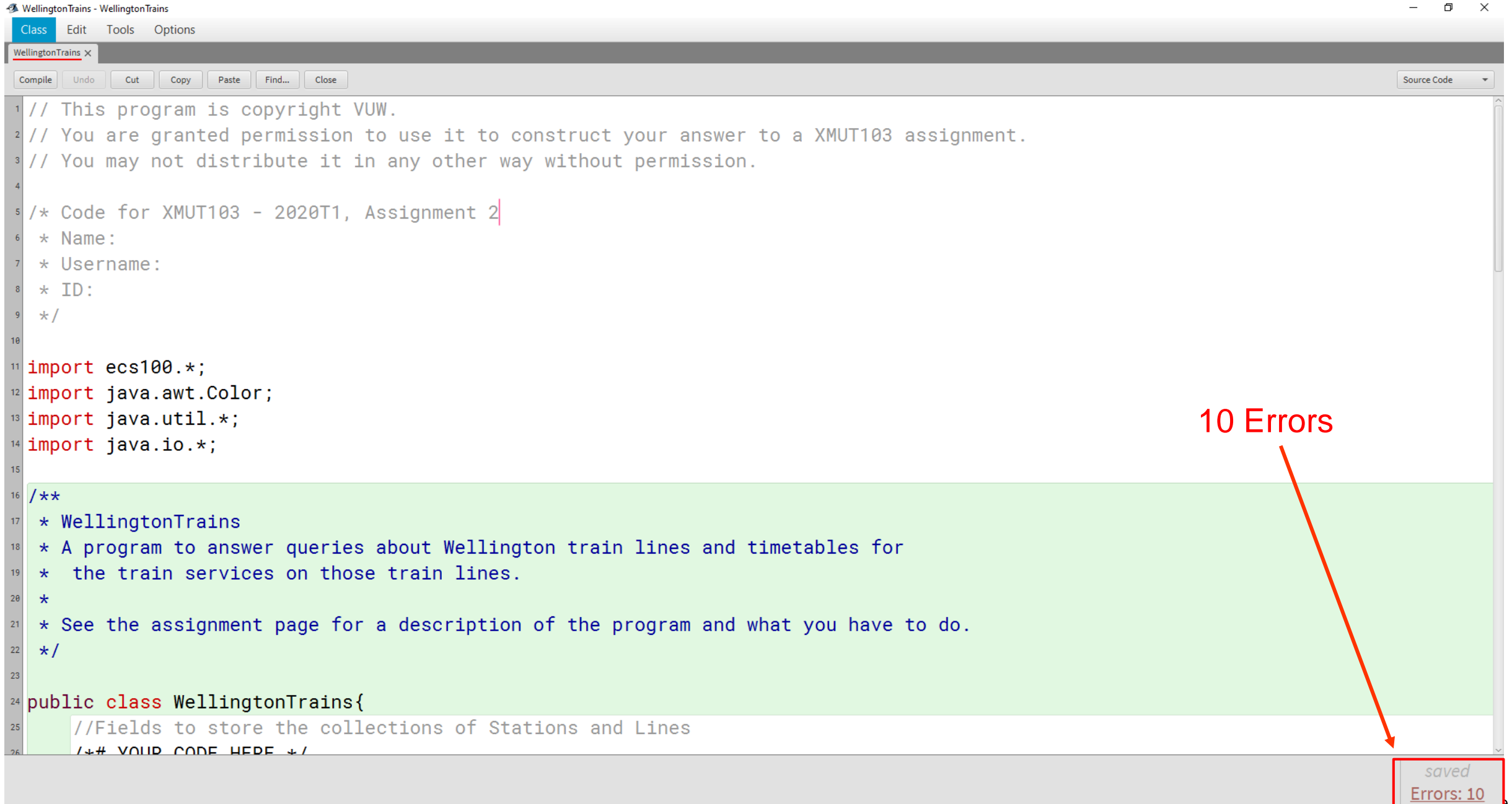
Victoria University of Wellington

https://ecs.wgtn.ac.nz/Courses/XMUT103_2020T1/Assignment2PartA

Name	Date modified	Type	Size
data	15/04/2020 3:51 PM	File folder	
package.bluej	15/04/2020 3:51 PM	BlueJ Project File	
Station.java	15/04/2020 3:51 PM	JAVA File	
TrainLine.java	15/04/2020 3:51 PM	JAVA File	
TrainService.java	15/04/2020 3:51 PM	JAVA File	
WellingtonTrains.java	15/04/2020 3:51 PM	JAVA File	

The screenshot shows the BlueJ IDE interface for a project named 'WellingtonTrains'. The main workspace displays four class icons: WellingtonTrains (top right, with a red checkered pattern), TrainLine (middle left, with diagonal stripes), Station (middle right, with diagonal stripes), and TrainService (bottom center, with diagonal stripes). A red arrow points from the 'WellingtonTrains.java' file in the file explorer on the left to the WellingtonTrains class icon in the workspace. The interface includes a menu bar (Project, Edit, Tools, View, Help), a toolbar with buttons for 'New Class...', 'Compile', 'Teamwork', and 'Testing', and a status bar at the bottom that reads 'Initialising virtual machine... Done.'.

WellingtonTrains.java file opened in BlueJ



WellingtonTrains - WellingtonTrains

Class Edit Tools Options

WellingtonTrains x

Compile Undo Cut Copy Paste Find... Close Source Code

```
1 // This program is copyright VUW.
2 // You are granted permission to use it to construct your answer to a XMUT103 assignment.
3 // You may not distribute it in any other way without permission.
4
5 /* Code for XMUT103 - 2020T1, Assignment 2
6  * Name:
7  * Username:
8  * ID:
9  */
10
11 import ecs100.*;
12 import java.awt.Color;
13 import java.util.*;
14 import java.io.*;
15
16 /**
17  * WellingtonTrains
18  * A program to answer queries about Wellington train lines and timetables for
19  * the train services on those train lines.
20  *
21  * See the assignment page for a description of the program and what you have to do.
22  */
23
24 public class WellingtonTrains{
25     //Fields to store the collections of Stations and Lines
26     /*# YOUR CODE HERE */
```

10 Errors

saved
Errors: 10

WellingtonTrains.java file opened in BlueJ

```
24 public class WellingtonTrains{
25     //Fields to store the collections of Stations and Lines
26     /*# YOUR CODE HERE */
27     ← private Map<String, Station> allStations = new HashMap<String, Station>();
28     // Fields for the suggested GUI.
29     private String stationName;           // station to get info about, or to start journey from
30     private String lineName;             // train line to get info about.
31     private String destinationName;
32     private int startTime = 0;           // time for enquiring about
33
34     /**
35     * main method: load the data and set up the user interface
36     */
37     public static void main(String[] args){
38         WellingtonTrains wel = new WellingtonTrains();
39         wel.loadData(); // load all the data
40         wel.setupGUI(); // set up the interface
41     }
```

WellingtonTrains.java file opened in BlueJ

```
43 /**
44  * Load data files
45  */
46 public void loadData(){
47     loadStationData();
48     UI.println("Loaded Stations");
49     loadTrainLineData();
50     UI.println("Loaded Train Lines");
51     // The following is only needed for the Completion and Challenge
52     loadTrainServicesData();
53     UI.println("Loaded Train Services");
54 }
```

Name	Date modified	Type	Size
fares.data	15/04/2020 3:51 PM	DATA File	1 KB
geographic-map.png	15/04/2020 3:51 PM	PNG File	91 KB
Johnsonville_Wellington-services.data	15/04/2020 3:51 PM	DATA File	2 KB
Johnsonville_Wellington-stations.data	15/04/2020 3:51 PM	DATA File	1 KB
Masterton_Wellington-services.data	15/04/2020 3:51 PM	DATA File	1 KB
Masterton_Wellington-stations.data	15/04/2020 3:51 PM	DATA File	1 KB
Melling_Wellington-services.data	15/04/2020 3:51 PM	DATA File	1 KB
Melling_Wellington-stations.data	15/04/2020 3:51 PM	DATA File	1 KB
stations.data	15/04/2020 3:51 PM	DATA File	1 KB
system-map.png	15/04/2020 3:51 PM	PNG File	70 KB
train-lines.data	15/04/2020 3:51 PM	DATA File	1 KB
Upper-Hutt_Wellington-services.data	15/04/2020 3:51 PM	DATA File	5 KB
Upper-Hutt_Wellington-stations.data	15/04/2020 3:51 PM	DATA File	1 KB
Waikanae_Wellington-services.data	15/04/2020 3:51 PM	DATA File	4 KB
Waikanae_Wellington-stations.data	15/04/2020 3:51 PM	DATA File	1 KB
Wellington_Johnsonville-services.data	15/04/2020 3:51 PM	DATA File	2 KB
Wellington_Johnsonville-stations.data	15/04/2020 3:51 PM	DATA File	1 KB
Wellington_Masterton-services.data	15/04/2020 3:51 PM	DATA File	1 KB
Wellington_Masterton-stations.data	15/04/2020 3:51 PM	DATA File	1 KB
Wellington_Melling-services.data	15/04/2020 3:51 PM	DATA File	1 KB
Wellington_Melling-stations.data	15/04/2020 3:51 PM	DATA File	1 KB

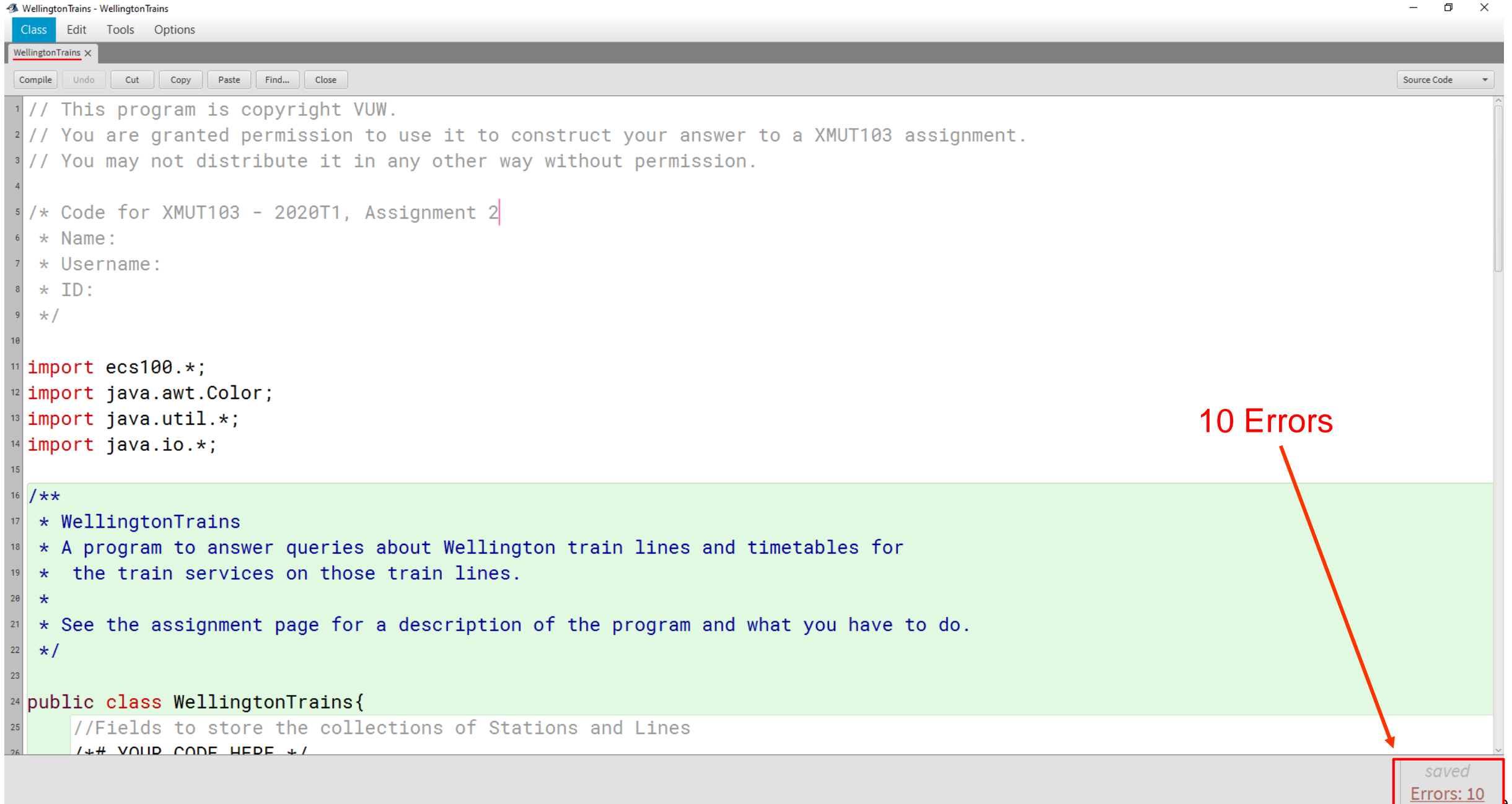
WellingtonTrains.java file opened in BlueJ

```
56 /**
57  * User interface has buttons for the queries and text fields to enter stations and train line
58  * You will need to implement the methods here.
59  */
60 public void setupGUI(){
61     UI.addButton("All Stations",    this::listAllStations);
62     UI.addButton("All Lines",      this::listAllTrainLines);
63     UI.addTextField("Station",      (String name) -> {this.stationName=name;});
64     UI.addTextField("Train Line",   (String name) -> {this.lineName=name;});
65     UI.addTextField("Destination", (String name) -> {this.destinationName=name;});
66     UI.addTextField("Time (24hr)", (String time) ->
67         {try{this.startTime=Integer.parseInt(time);}catch(Exception e){UI.println("Enter four digits");}});
68     UI.addButton("Lines of Station",    () -> {listLinesOfStation(this.stationName);});
69     UI.addButton("Stations on Line",    () -> {listStationsOnLine(this.lineName);});
70     UI.addButton("Stations connected?", () -> {checkConnected(this.stationName, this.destinationName);});
71     UI.addButton("Next Services",      () -> {findNextServices(this.stationName, this.startTime);});
72     UI.addButton("Find Trip",          () -> {findTrip(this.stationName, this.destinationName, this.startTime);});
73     UI.addButton("Quit", UI::quit);
74     UI.setDivider(1.0);
75 }
```

WellingtonTrains.java file opened in BlueJ

```
77 // Methods for loading data and answering queries
78
79 /*# YOUR CODE HERE */
80
81 // Utility method to help with reading data files
82 /** Read all lines in a file into a list of Strings */
83 public ArrayList<String> readAllLines(String filename){
84     try{
85         ArrayList<String> ans = new ArrayList<String>();
86         Scanner sc = new Scanner(new File(filename));
87         while (sc.hasNext()){
88             ans.add(sc.nextLine());
89         }
90         sc.close();
91         return ans;
92     }
93     catch(IOException e){UI.println("Fail: " + e); return null;}
94 }
95
96 }
```

WellingtonTrains.java file opened in BlueJ



```
1 // This program is copyright VUW.
2 // You are granted permission to use it to construct your answer to a XMUT103 assignment.
3 // You may not distribute it in any other way without permission.
4
5 /* Code for XMUT103 - 2020T1, Assignment 2
6  * Name:
7  * Username:
8  * ID:
9  */
10
11 import ecs100.*;
12 import java.awt.Color;
13 import java.util.*;
14 import java.io.*;
15
16 /**
17  * WellingtonTrains
18  * A program to answer queries about Wellington train lines and timetables for
19  * the train services on those train lines.
20  *
21  * See the assignment page for a description of the program and what you have to do.
22  */
23
24 public class WellingtonTrains{
25     //Fields to store the collections of Stations and Lines
26     /*# YOUR CODE HERE */
```

10 Errors

saved
Errors: 10

WellingtonTrains.java file opened in BlueJ

```
43  /**
44   * Load data files
45   */
46  public void loadData(){
47      loadStationData();
48      UI.println("Loaded Stations");
49      loadTrainLineData();
50      UI.println("Loaded Train Lines");
51      // The following is only needed for the Completion and Challenge
52      loadTrainServicesData();
53      UI.println("Loaded Train Services");
54  }
```

1) Before

3) After

2) Insert/type this method

```
46  public void loadData(){
47      loadStationData();
48      UI.println("Loaded Stations");
49      loadTrainLineData();
50      UI.println("Loaded Train Lines");
51      // The following is only needed for the Completion and Challenge
52      loadTrainServicesData();
53      UI.println("Loaded Train Services");
54  }
55
56  public void loadStationData() {
57  }
```

WellingtonTrains.java file opened in BlueJ

```
46 public void loadData(){
47     loadStationData();
48     UI.println("Loaded Stations");
49     2 loadTrainLineData();
50     UI.println("Loaded Train Lines");
51     // The following is only needed for the Completion and Challenge
52     3 loadTrainServicesData();
53     UI.println("Loaded Train Services");
54 }
```

```
56 public void loadStationData() {
57 }
```

```
59 public void loadTrainLineData() {
60 }
```

```
62 public void loadTrainServicesData() {
63 }
```

2

3

WellingtonTrains.java file opened in BlueJ

```

65  /**
66   * User interface has buttons for the queries and text fields to enter stations and train line
67   * You will need to implement the methods here.
68   */
69  public void setupGUI(){
70      UI.addButton("All Stations", this::listAllStations);
71      UI.addButton("All Lines", this::listAllTrainLines);

```

1) Before

3) After

2) Insert this method

```

69  public void setupGUI(){
70      UI.addButton("All Stations", this::listAllStations);
71      UI.addButton("All Lines", this::listAllTrainLines);
72      UI.addTextField("Station", (String name) -> {this.stationName=name;});
73      UI.addTextField("Train Line", (String name) -> {this.lineName=name;});
74      UI.addTextField("Destination", (String name) -> {this.destinationName=name;});
75      UI.addTextField("Time (24hr)", (String time) ->
76          {try{this.startTime=Integer.parseInt(time);}catch(Exception e){UI.println("Enter four digits");}});
77      UI.addButton("Lines of Station", () -> {listLinesOfStation(this.stationName);});
78      UI.addButton("Stations on Line", () -> {listStationsOnLine(this.lineName);});
79      UI.addButton("Stations connected?", () -> {checkConnected(this.stationName, this.destinationName);});
80      UI.addButton("Next Services", () -> {findNextServices(this.stationName, this.startTime);});
81      UI.addButton("Find Trip", () -> {findTrip(this.stationName, this.destinationName, this.startTime);});
82      UI.addButton("Quit", UI::quit);
83      UI.setDivider(1.0);
84  }
85
86  // Methods for loading data and answering queries
87
88  /*# YOUR CODE HERE */
89  public void listAllStations() {
90  }

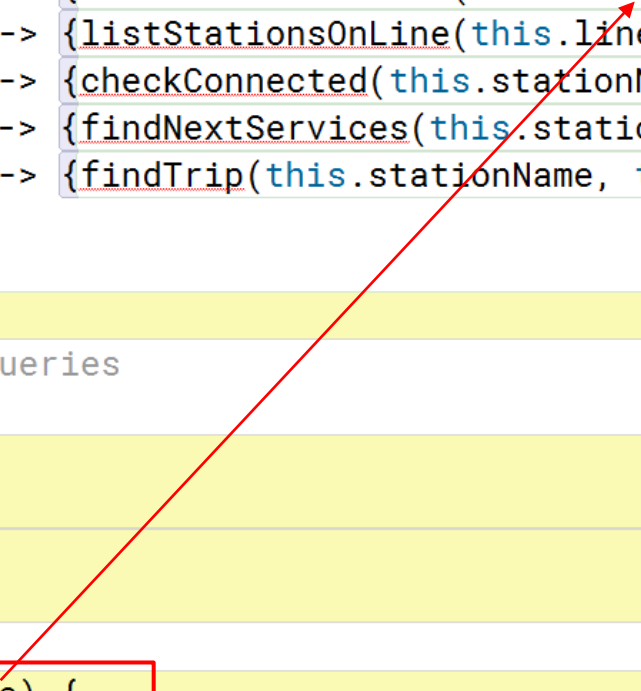
```

WellingtonTrains.java file opened in BlueJ

```
69 public void setupGUI(){
70     UI.addButton("All Stations",    this::listAllStations);
71     UI.addButton("All Lines",      this::listAllTrainLines);
72     UI.addTextField("Station",     (String name) -> {this.stationName=name;});
73     UI.addTextField("Train Line",  (String name) -> {this.lineName=name;});
74     UI.addTextField("Destination", (String name) -> {this.destinationName=name;});
75     UI.addTextField("Time (24hr)", (String time) ->
76         {try{this.startTime=Integer.parseInt(time);}catch(Exception e){UI.println("Enter four digits");}});
77     UI.addButton("Lines of Station",    () -> {listLinesOfStation(this.stationName);});
78     UI.addButton("Stations on Line",    () -> {listStationsOnLine(this.lineName);});
79     UI.addButton("Stations connected?", () -> {checkConnected(this.stationName, this.destinationName);});
80     UI.addButton("Next Services",      () -> {findNextServices(this.stationName, this.startTime);});
81     UI.addButton("Find Trip",          () -> {findTrip(this.stationName, this.destinationName, this.startTime);});
82     UI.addButton("Quit", UI::quit);
83     UI.setDivider(1.0);
84 }
85 // Methods for loading data and answering queries
86 /*# YOUR CODE HERE */
87 public void listAllStations() {
88 }
89 public void listAllTrainLines() {
90 }
91 // Utility method to help with reading data files
92 /** Read all lines in a file into a list of Strings */
93 public ArrayList<String> readAllLines(String filename){
```

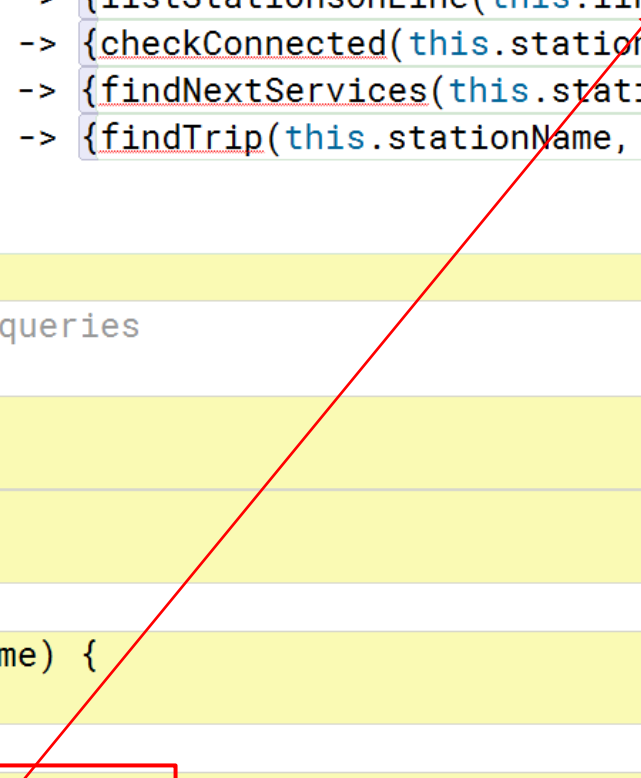
WellingtonTrains.java file opened in BlueJ

```
75 UI.addTextField("Time (24hr)", (String time) ->
76     {try{this.startTime=Integer.parseInt(time);}catch(Exception e){UI.println("Enter four digits");}});
77 UI.addButton("Lines of Station",      () -> {listLinesOfStation(this.stationName);});
78 UI.addButton("Stations on Line",     () -> {listStationsOnLine(this.lineName);});
79 UI.addButton("Stations connected?",  () -> {checkConnected(this.stationName, this.destinationName);});
80 UI.addButton("Next Services",        () -> {findNextServices(this.stationName, this.startTime);});
81 UI.addButton("Find Trip",            () -> {findTrip(this.stationName, this.destinationName, this.startTime);});
82 UI.addButton("Quit", UI::quit);
83 UI.setDivider(1.0);
84 }
85 // Methods for loading data and answering queries
86 /*# YOUR CODE HERE */
87 public void listAllStations() {
88 }
89 public void listAllTrainLines() {
90 }
91
92 public void listLinesOfStation(String stName) {
93 }
```



WellingtonTrains.java file opened in BlueJ

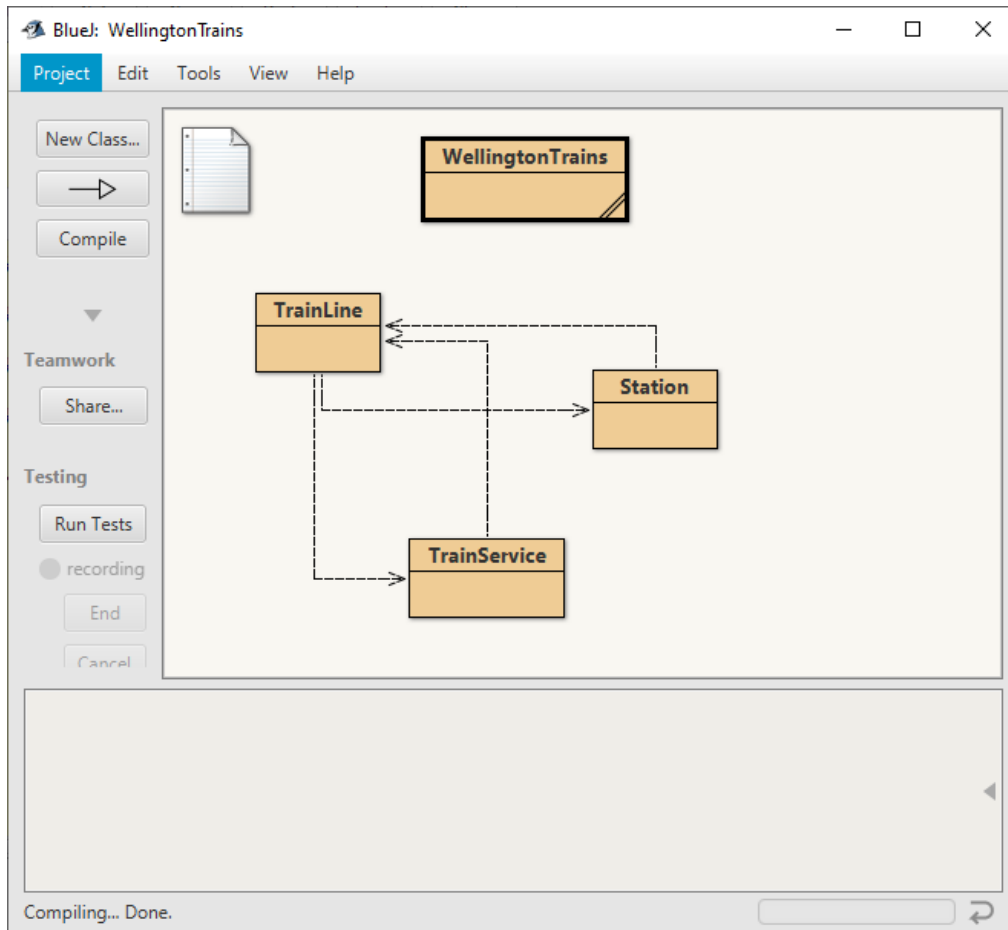
```
76     {try{this.startTime=Integer.parseInt(time);}catch(Exception e){UI.println("Enter four digits");}});
77     UI.addButton("Lines of Station",    () -> {listLinesOfStation(this.stationName);});
78     UI.addButton("Stations on Line",    () -> {listStationsOnLine(this.lineName);});
79     UI.addButton("Stations connected?", () -> {checkConnected(this.stationName, this.destinationName);});
80     UI.addButton("Next Services",       () -> {findNextServices(this.stationName, this.startTime);});
81     UI.addButton("Find Trip",           () -> {findTrip(this.stationName, this.destinationName, this.startTime);});
82     UI.addButton("Quit", UI::quit);
83     UI.setDivider(1.0);
84 }
85 // Methods for loading data and answering queries
86 /*# YOUR CODE HERE */
87 public void listAllStations() {
88 }
89 public void listAllTrainLines() {
90 }
91
92 public void listLinesOfStation(String stName) {
93 }
94
95 public void listStationsOnLine(String lnName) {
96 }
```



WellingtonTrains.java file opened in BlueJ

```
WellingtonTrains - WellingtonTrains
Class Edit Tools Options
WellingtonTrains X
Compile Undo Cut Copy Paste Find... Close
84 }
85 // Methods for loading data and answering queries
86 /*# YOUR CODE HERE */
87 public void listAllStations() {
88 }
89 public void listAllTrainLines() {
90 }
91
92 public void listLinesOfStation(String stName) {
93 }
94 public void listStationsOnLine(String lnName) {
95 }
96
97 public void checkConnected(String stName, String desName) {
98 }
99
100 public void findNextServices(String stName, int stTime) {
101 }
102
103 public void findTrip(String stName, String desName, int stTime) {
104 }
```

Compile & run the WellingtonTrains program



BlueJ: WellingtonTrains

Project Edit Tools View Help

New Class...

Compile

Teamwork

Share...

Testing

Run Tests

recording

End

Cancel

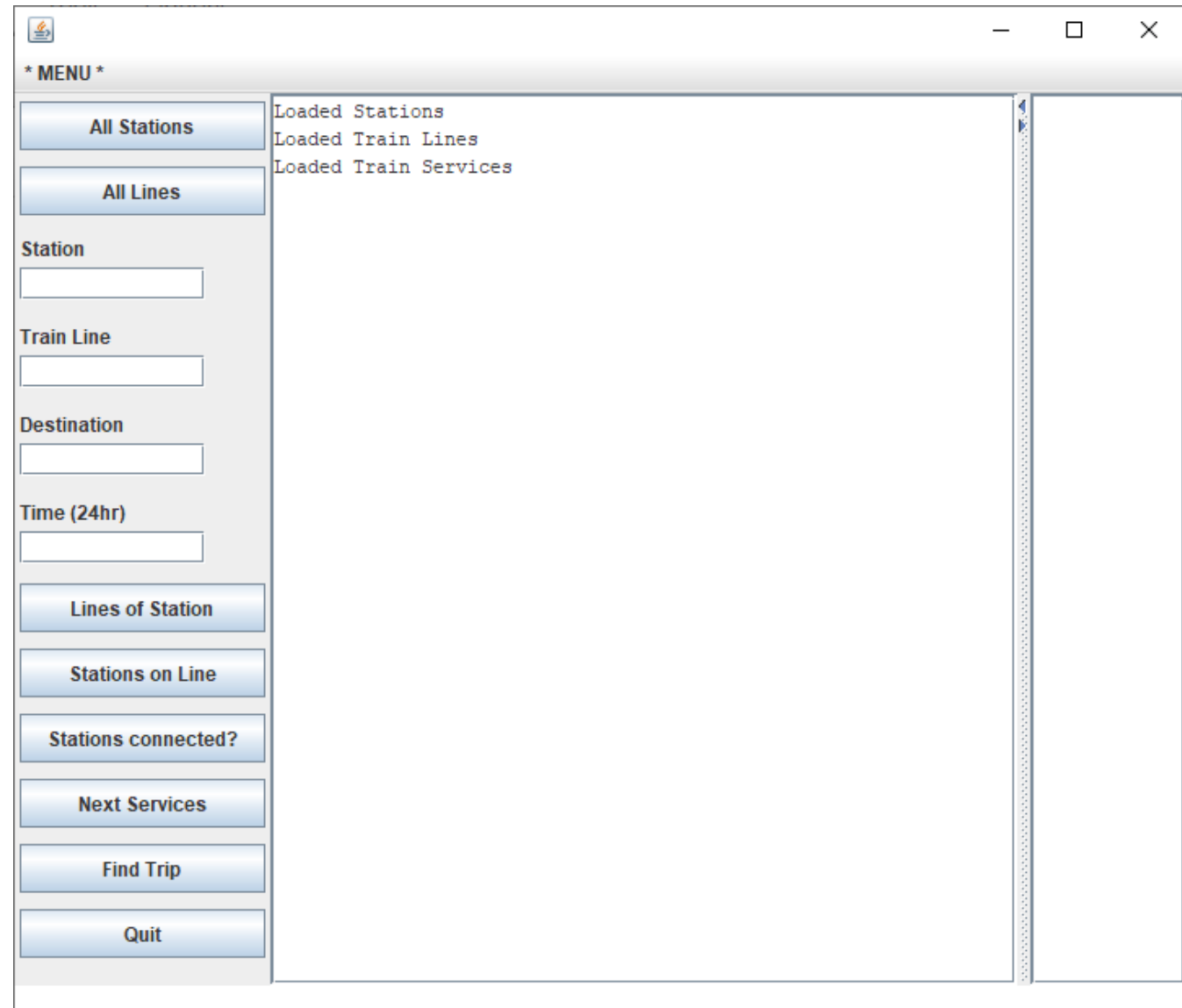
WellingtonTrains

TrainLine

Station

TrainService

Compiling... Done.



* MENU *

All Stations

All Lines

Station

Train Line

Destination

Time (24hr)

Lines of Station

Stations on Line

Stations connected?

Next Services

Find Trip

Quit

Loaded Stations

Loaded Train Lines

Loaded Train Services