School of Computing and Information Technology
University of Wollongong

Session: **Autumn 2024**
Lecturer: **Wei Zong**

CSIT213 Autumn Session 2024
Assignment 2
Published on 24 March 2024 **Due:** Saturday 04 May, 11:30 pm
Total Marks: 15 marks

Scope

This assignment is related to the UML classes diagrams, Java classes (include interface and abstract classes) definitions and implementations, polymorphism, collectors, and file input/output.

Please read very carefully the information listed below.

General Java coding requirements:

- Create your programs with good programming style and form using proper blank spaces, indentation, and braces to make your code easy to read and understand.
- Create identifiers with sensible names.
- Add proper comments to describe your code segments where they are necessary for readers to understand what your code intends to achieve.
- Logical structures and statements are properly used for specific purposes.
- Read the assignment specification carefully, and make sure that you follow the direction in this assignment. In **every assignment source file** that you will submit on this subject, you must put the following information in the header of your program:

```
/*-------------------------------------------------------
   My name:
   My student number:
   My course code: CSIT213
   My email address:
   Assignment number: 2
-------------------------------------------------------*/
```

A submission procedure is explained at the end of the specification.

It is recommended to solve the problems before attending the laboratory classes in order to efficiently use supervised laboratory time.

A submission marked by Moodle as `Late` is treated as a late submission no matter how many seconds it is late.

A policy regarding late submissions is included in the subject outline.

A submission of compressed files (zipped, gzipped, rared, tared, 7-zipped, lhzed, … etc) is **not allowed**. The compressed files will not be evaluated.

An implementation that does not compile due to one or more syntactical or processing errors scores no marks. It is expected that all tasks included in **Assignment 2** will be solved **individually without any cooperation** from the other students. If you have any doubts, questions, etc. please consult your lecturer or tutor during lab classes or office hours. Plagiarism will result in a **FAIL** grade being recorded for the assessment task.

Tasks

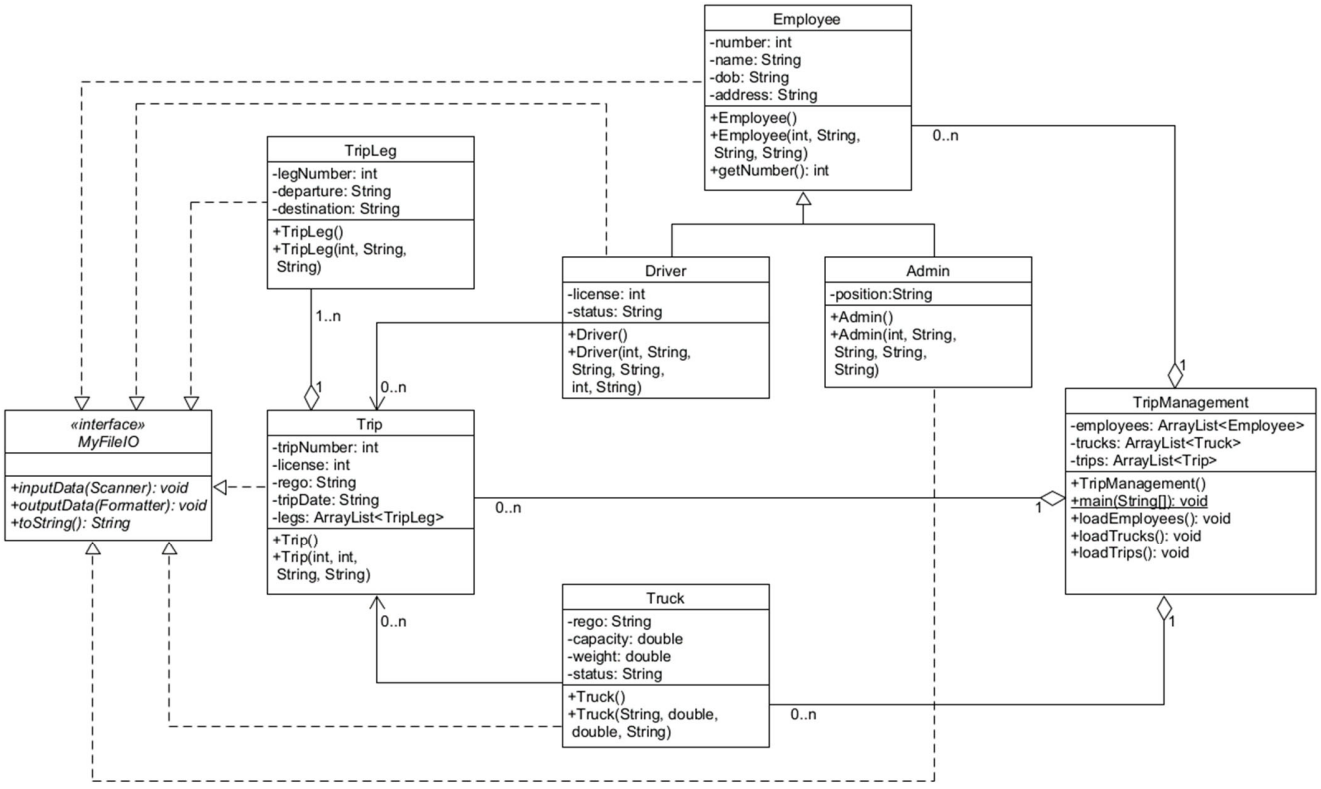
In this assignment, you are required to design and implement a Trip Management System (TMS) in Java. This system helps a company to manage employees, trucks, and trips. The related topics include interface, abstract, polymorphism, and file I/O.

Implementation

The TMS system helps a company to manage employees, trucks, and trips. The system loads data of employees, trucks, and trips from text files when the program starts. Then the program displays a menu so that a user can choose what to do.

A user can choose to display information about all employees, trucks, or trips. A user can find an employee, a truck, or a trip. A user can add a new employee, a new truck, or a new trip information. A user can save all the data into text files.

The UML class diagram of TMS is given below. You can add new classes, methods and attributes in the UML class diagram but CANNOT modify or delete any existing classes, attributes, and methods. Your java implementation must be consistent with the UML class diagram.



First, we define an interface class `MyFileIO` in a file `MyFileIO.java` that only consists of three abstract methods: `inputData(Scanner)`, `outputData(Formatter)` and `toString()`.

Define a superclass `Employee` and two sub-classes `Admin` and `Driver` in a file `Employee.java`. The class Employee implements the interface MyFileIO. Two sub-classes override the methods defined in the interface `MyFileIO`.

Define a class `Truck` in a file `Truck.java` that implements the interface `MyFileIO`.

Define a class `Trip` in a file `Trip.java` that implements the interface `MyFileIO` and contains ArrayList container containing objects of `TripLeg`.

Define a class `TripLeg` in a file `Trip.java` that implements the interface `MyFileIO`.

The `TripManagement` class is defined in a file `TripManagement.java` that contains `ArrayList` containers containing objects of `Employee`, `Truck` and `Trip`.

The application initially loads data of employees, truck, and trips from text files `employees.txt`, `trucks.txt` and `trips.txt` and stores them in the containers by calling the methods `loadEmployees()`, `loadTrucks()` and `loadTrips()`.

The format of a file `employees.txt` contains data like

D, 1, John Smith, 02-12-1971, 42 Victoria St. Hurstville NSW 2456, 10001, AVAILABLE

D, 2, Peter Taylor, 01-12-1970, 42 Victoria St. Hurstville NSW 2456, 10008, ON LEAVE

A, 9, John Lucas, 12-16-1966, 20 Huxley St. Horsley NSW 2530, SUPPORT

A, 10, John Fox, 10-15-1975, 18 Victoria St. Hurstville NSW 2456, DIRECTOR

D, 11, Adam Fox, 08-10-1974, 45 Victoria St. Hurstville NSW 2456, 30005, BUSY

Each row is a record of an employee. The first character is the type of employee. The letter `D` means it is a Driver record. The letter `A` means it is an Admin record. Each field is separated by a comma(,) and a space ( ).

An admin record contains an employee number, name, date of birth, address, and position.

A driver record contains an employee number, name, date of birth, address, license number, and status.

The format of a file `trucks.txt` contains data like

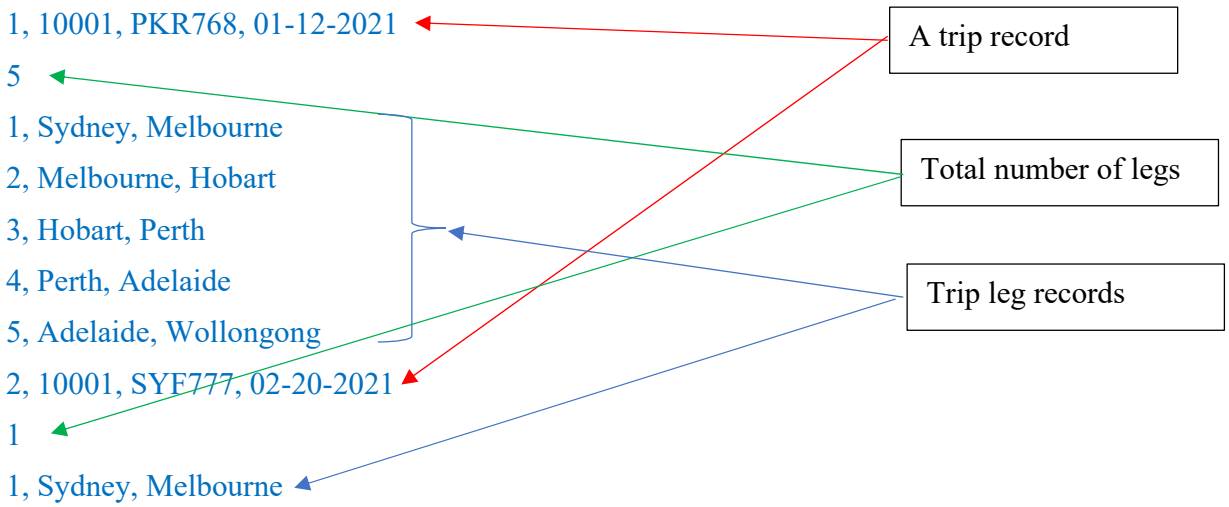
PKR768, 1234.50, 3000.20, AVAILABLE

SST005, 12000.30, 50000.10, USED

QRT834, 5550.40, 400.50, USED

Each row is a record of a truck. A truck record contains a registration number, capacity, weight, and status. Each field is separated by a comma (,) and a space ( ).

The format of a file trips.txt contains data like



A trip record consists of a trip number, a driver's license number, a truck registration number, a trip date and followed by a total number of trip legs and 1 or more trip leg records accordingly.

Each trip leg consists of a leg number, a departure city, and a destination city. Each field is separated by a comma (,) and a space ( ).

Hint: You can open a text file, and then use the method

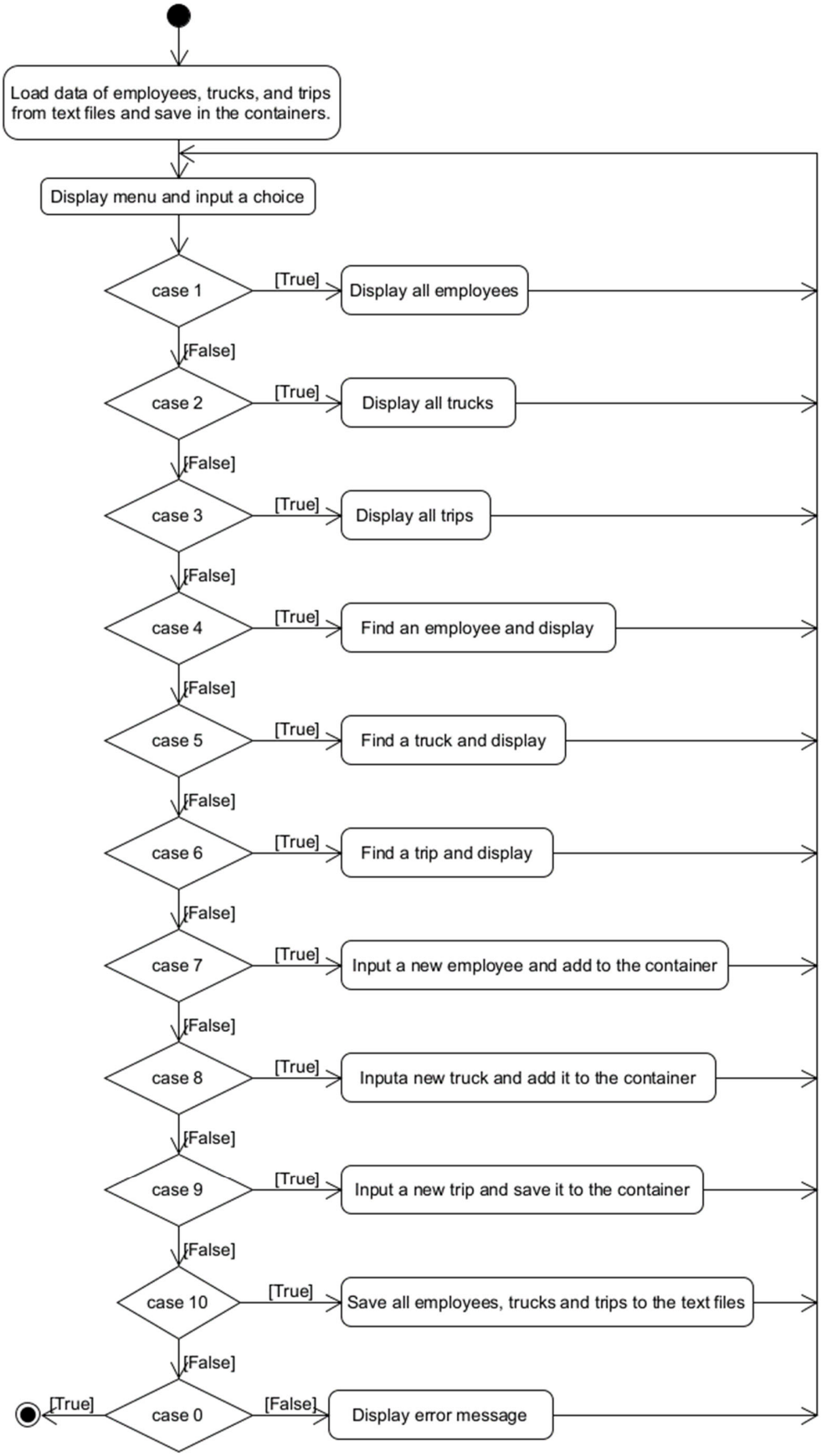
```
useDelimiter(", |\r\n|\n")
```

of a `Scanner` object before getting input data from a text file.

After loading the data into the containers, the program displays menu and ask a user to choose an item until the input value is 0.

- If an input value is 1, the program calls a method to display all employees' information from the container `employees`.
- If an input value is 2, the program calls a method to display all trucks' information from the container `trucks`.
- If an input value is 3, the program calls a method to display all trips' information from the container `trips`.
- If an input value is 4, the program calls a method to get an input employee number, find the employee and display its information.
- If an input value is 5, the program calls a method to get an input truck registration number, find the truck and display its information.
- If an input value is 6, the program calls a method to get an input trip number, find the trip and display its information.
- If an input value is 7, the program calls a method to get inputs of an employee. If the employee does not exist, store the new employee object in the container `employees`.
- If an input value is 8, the program calls a method to get inputs of a truck. If the truck does not exist, store the truck object in the container `trucks`.
- If an input value is 9, the program calls a method to get inputs of a trip (including trip legs of the trip). If the trip does not exist, store the trip object in the container `trips`.
- If an input value is 10, the program calls methods to save all the data from containers into the text files. The format of each text file must be the same as the test files `employees.txt`, `trucks.txt`, and `trips.txt` like the examples above.
- If an input value is 0, the program displays the message `Bye-bye`, and then exits.

UML activity diagram



Testing files

Download files `employees.txt`, `trucks.txt` and `trips.txt` from the Moodle site.

Compilation and testing

Compile your program by using the `javac` command.

```
javac TripManagement.java
```

Process your program by using the `java` command.

```
java TripManagement
```

Test your program for all the items. See the examples of the processing results below for more details.

Processing example

Examples of application processing are given below. The user's inputs are highlighted in red colour.

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.

Input a choice (0-10): 1

Driver Employee number: 1, Employee name: John Smith, Date of birth: 02-12-1971, Address: 42 Victoria St. Hurstville NSW 2456, License: 10001, Status: AVAILABLE
Driver Employee number: 2, Employee name: Peter Taylor, Date of birth: 01-12-1970, Address: 42 Victoria St. Hurstville NSW 2456, License: 10008, Status: ON LEAVE
Driver Employee number: 3, Employee name: John Doe, Date of birth: 03-23-1966, Address: 12 Station St. Dapto NSW 2530, License: 10002, Status: AVAILABLE
Driver Employee number: 4, Employee name: John Gray, Date of birth: 05-05-1988, Address: 16 Station St. Dapto NSW 2530, License: 10004, Status: AVAILABLE
Driver Employee number: 5, Employee name: Adam Taylor, Date of birth: 01-01-1980, Address: 42 Church St. City NSW 2300, License: 10003, Status: ON LEAVE
Driver Employee number: 6, Employee name: Michael Jones, Date of birth: 03-05-1975, Address: 23 Waterloo Ave. Surry Hills NSW 2502, License: 10012, Status: AVAILABLE
Driver Employee number: 7, Employee name: Frederic Jones, Date of birth: 02-10-1978, Address: 3 Victoria St. Redfern NSW 2420, License: 20002, Status: BUSY
Driver Employee number: 8, Employee name: Peter O'Brien, Date of birth: 02-28-1983, Address: 19 Lucas Dr. Horsley NSW 2530, License: 20003, Status: 'BUSY'
Admin Employee number: 9, Employee name: John Lucas, Date of birth: 12-16-1966, Address: 20 Huxley St. Horsley NSW 2530, Position: SUPPORT
Admin Employee number: 10, Employee name: John Fox, Date of birth: 10-15-1975, Address: 18 Victoria St. Hurstville NSW 2456, Position: DIRECTOR
Driver Employee number: 11, Employee name: Adam Fox, Date of birth: 08-10-1974, Address: 45 Victoria St. Hurstville NSW 2456, License: 30005, Status: BUSY

Driver Employee number: 12, Employee name: Phillip Cox, Date of birth: 12-12-1985, Address: 5 The Avenue Rockdale NSW 2300, License: 40002, Status: BUSY
Driver Employee number: 13, Employee name: Andrew K Smith, Date of birth: 04-04-1969, Address: 42 Bambaramba Ave. Pennant Hills NSW 2556, License: 20045, Status: AVAILABLE
Admin Employee number: 14, Employee name: Andrew R Smith, Date of birth: 04-01-1992, Address: 67 King Cr. Hurstville NSW 2456, Position: CEO
Admin Employee number: 15, Employee name: Michael Potter, Date of birth: 04-01-1995, Address: 568 Bong Bong St. Horsley NSW 2530, Position: SUPPORT
Driver Employee number: 16, Employee name: Harry Potter, Date of birth: 04-01-1987, Address: 568 Bong Bong St. Horsley NSW 2530, License: 20055, Status: AVAILABLE
Driver Employee number: 17, Employee name: James Bond, Date of birth: 02-14-1989, Address: 7 Alan Bond St. Perth WA 6000, License: 20065, Status: AVAILABLE
Driver Employee number: 18, Employee name: Paris Hilton, Date of birth: 05-01-1977, Address: 1 Hilton St. Melbourne VIC 3000, License: 10305, Status: AVAILABLE
Admin Employee number: 19, Employee name: Lady Gaga, Date of birth: 06-01-1992, Address: 3 Pork st. Hobart TAS 7000, Position: SUPPORT
Driver Employee number: 20, Employee name: Robin Hood, Date of birth: 05-23-1999, Address: 6 Nottingham Pl. Sydney NSW 2000, License: 10345, Status: AVAILABLE

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10):2
Rego: PKR768, Capacity: 1234.50, Weight: 3000.20, Status: AVAILABLE
Rego: SST005, Capacity: 12000.30, Weight: 50000.10, Status: USED
Rego: QRT834, Capacity: 5550.40, Weight: 400.50, Status: USED
Rego: LUCY01, Capacity: 3000.20, Weight: 1000.20, Status: AVAILABLE
Rego: KKK007, Capacity: 10000.50, Weight: 3000.40, Status: MAINTAINED
Rego: SYF777, Capacity: 3333.60, Weight: 4566.50, Status: MAINTAINED
Rego: PKR008, Capacity: 22000.30, Weight: 8800.50, Status: AVAILABLE
Rego: XCF003, Capacity: 30000.10, Weight: 10000.20, Status: AVAILABLE
Rego: GFT008, Capacity: 40000.30, Weight: 15000.40, Status: AVAILABLE
Rego: LUCY02, Capacity: 43000.20, Weight: 3000.50, Status: AVAILABLE
Rego: AL08UK, Capacity: 50000.10, Weight: 5000.00, Status: AVAILABLE

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.

10. Save all data into files.
0. Exit.
Input a choice (0-10): 3
Trip number: 1, license: 10001, Rego: PKR768, Trip date: 01-12-2021
Leg number: 1, Departure: Sydney, Destination: Melbourne
Leg number: 2, Departure: Melbourne, Destination: Hobart
Leg number: 3, Departure: Hobart, Destination: Perth
Leg number: 4, Departure: Perth, Destination: Adelaide
Leg number: 5, Departure: Adelaide, Destination: Wollongong

Trip number: 2, license: 10001, Rego: SYF777, Trip date: 02-20-2021
Leg number: 1, Departure: Sydney, Destination: Melbourne

Trip number: 3, license: 10001, Rego: KKK007, Trip date: 03-12-2021
Leg number: 1, Departure: Sydney, Destination: Melbourne

Trip number: 4, license: 10001, Rego: PKR768, Trip date: 06-29-2021
Leg number: 1, Departure: Sydney, Destination: Melbourne

Trip number: 5, license: 20002, Rego: PKR768, Trip date: 01-12-2021
Leg number: 1, Departure: Melbourne, Destination: Sydney

Trip number: 6, license: 10002, Rego: SYF777, Trip date: 02-20-2021
Leg number: 1, Departure: Melbourne, Destination: Sydney

Trip number: 7, license: 30005, Rego: KKK007, Trip date: 03-12-2021
Leg number: 1, Departure: Melbourne, Destination: Sydney

Trip number: 8, license: 10001, Rego: PKR768, Trip date: 01-13-2021
Leg number: 1, Departure: Sydney, Destination: Newcastle
Leg number: 2, Departure: Newcastle, Destination: Brisbane
Leg number: 3, Departure: Brisbane, Destination: Perth

Trip number: 9, license: 10002, Rego: QRT834, Trip date: 09-17-2021
Leg number: 1, Departure: Sydney, Destination: Melbourne

Trip number: 10, license: 30005, Rego: KKK007, Trip date: 12-15-2022
Leg number: 1, Departure: Sydney, Destination: Wollongong

Trip number: 11, license: 10003, Rego: SST005, Trip date: 01-23-2022
Leg number: 1, Departure: Melbourne, Destination: Sydney

Trip number: 12, license: 10002, Rego: PKR768, Trip date: 03-12-2022
Leg number: 1, Departure: Melbourne, Destination: Sydney

Trip number: 13, license: 20002, Rego: QRT834, Trip date: 04-23-2022
Leg number: 1, Departure: Sydney, Destination: Melbourne
Leg number: 2, Departure: Melbourne, Destination: Sydney

Trip number: 14, license: 20002, Rego: PKR008, Trip date: 04-23-2022
Leg number: 1, Departure: Wollongong, Destination: Sydney

Trip number: 15, license: 30005, Rego: PKR768, Trip date: 05-24-2022
Leg number: 1, Departure: Sydney, Destination: Melbourne

Trip number: 16, license: 30005, Rego: SST005, Trip date: 08-02-2022
Leg number: 1, Departure: Sydney, Destination: Wollongong

Trip number: 17, license: 20002, Rego: QRT834, Trip date: 09-17-2022
Leg number: 1, Departure: Sydney, Destination: Wollongong
Leg number: 2, Departure: Wollongong, Destination: Sydney

Trip number: 18, license: 10001, Rego: KKK007, Trip date: 12-15-2022
Leg number: 1, Departure: Melbourne, Destination: Sydney

Trip number: 19, license: 30005, Rego: SST005, Trip date: 01-23-2022
Leg number: 1, Departure: Melbourne, Destination: Sydney

Trip number: 20, license: 10003, Rego: PKR768, Trip date: 03-12-2022
Leg number: 1, Departure: Sydney, Destination: Melbourne

Trip number: 21, license: 10001, Rego: QRT834, Trip date: 04-23-2022
Leg number: 1, Departure: Wollongong, Destination: Sydney

Trip number: 22, license: 30005, Rego: PKR008, Trip date: 04-23-2022
Leg number: 1, Departure: Melbourne, Destination: Sydney

Trip number: 23, license: 10003, Rego: PKR768, Trip date: 05-25-2022
Leg number: 1, Departure: Wollongong, Destination: Sydney

Trip number: 24, license: 20002, Rego: SST005, Trip date: 08-02-2022

Leg number: 1, Departure: Sydney, Destination: Melbourne

Trip number: 25, license: 10001, Rego: PKR768, Trip date: 01-12-2023
Leg number: 1, Departure: Melbourne, Destination: Sydney
Leg number: 2, Departure: Sydney, Destination: Perth
Leg number: 3, Departure: Perth, Destination: Sydney
Leg number: 4, Departure: Sydney, Destination: Brisbane

Trip number: 26, license: 10001, Rego: SYF777, Trip date: 02-20-2023
Leg number: 1, Departure: Brisbane, Destination: Sydney

Trip number: 27, license: 20002, Rego: KKK007, Trip date: 03-12-2023
Leg number: 1, Departure: Sydney, Destination: Wollongong

Trip number: 28, license: 30005, Rego: PKR768, Trip date: 06-29-2023
Leg number: 1, Departure: Sydney, Destination: Melbourne
Leg number: 2, Departure: Melbourne, Destination: Perth

Trip number: 29, license: 10001, Rego: QRT834, Trip date: 06-17-2023
Leg number: 1, Departure: Sydney, Destination: Perth

Trip number: 30, license: 10002, Rego: KKK007, Trip date: 06-15-2023
Leg number: 1, Departure: Melbourne, Destination: Sydney
Leg number: 2, Departure: Sydney, Destination: Melbourne

Trip number: 31, license: 10003, Rego: SST005, Trip date: 01-23-2023
Leg number: 1, Departure: Sydney, Destination: Melbourne

Trip number: 32, license: 20002, Rego: PKR768, Trip date: 03-12-2023
Leg number: 1, Departure: Sydney, Destination: Melbourne

Trip number: 33, license: 30005, Rego: QRT834, Trip date: 04-23-2023
Leg number: 1, Departure: Sydney, Destination: Melbourne

Trip number: 34, license: 30005, Rego: PKR008, Trip date: 04-23-2023
Leg number: 1, Departure: Wollongong, Destination: Sydney

Trip number: 35, license: 10001, Rego: PKR768, Trip date: 05-24-2023
Leg number: 1, Departure: Sydney, Destination: Melbourne
Leg number: 2, Departure: Melbourne, Destination: Adelaide
Leg number: 3, Departure: Adelaide, Destination: Perth

Leg number: 4, Departure: Perth, Destination: Sydney

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 4
Employee number: 10
Admin Employee number: 10, Employee name: John Fox, Date of birth: 10-15-1975, Address: 18 Victoria St. Hurstville NSW 2456, Position: DIRECTOR

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 4
Employee number: 25
Employee 25 does not exist

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 5
Truck rego: ABC001
Truck ABC001 does not exist

1. Display all employees.
2. Display all trucks.

3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 5
Truck rego: PKR008
Truck PKR008 exists
Rego: PKR008, Capacity: 22000.30, Weight: 8800.50, Status: AVAILABLE

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 6
Trip number: 50
Trip 50 does not exist

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 6
Trip number: 35
Trip number: 35, license: 10001, Rego: PKR768, Trip date: 05-24-2023
Leg number: 1, Departure: Sydney, Destination: Melbourne
Leg number: 2, Departure: Melbourne, Destination: Adelaide
Leg number: 3, Departure: Adelaide, Destination: Perth
Leg number: 4, Departure: Perth, Destination: Sydney

1. Display all employees.
2. Display all trucks.
3. Display all trips.

4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 7
Employee number: 10
The employee 10 exists.

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 7
Employee number: 21
Employee name: James Bond
Data of birth (dd-mm-yyyy): 10-03-1992
Address: 123 Princess Street Wollongong NSW 2500
Admin or Driver (A or D): D
License: 123456
Status: AVAILABLE

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 7
Employee number: 22
Employee name: Albert Einstein
Data of birth (dd-mm-yyyy): 08-02-1990
Address: 20 Physicist road Wollongong NSW 2500
Admin or Driver (A or D): A
Position: SUPPORT

1. Display all employees.

2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 4
Employee number: 21
Driver Employee number: 21, Employee name: James Bond, Date of birth: 10-03-1992, Address: 123 Princess Street Wollongong NSW 2500, License: 123456, Status: AVAILABLE

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 4
Employee number: 22
Admin Employee number: 22, Employee name: Albert Einstein, Date of birth: 08-02-1990, Address: 20 Physicist road Wollongong NSW 2500, Position: SUPPORT

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 8
Truck rego: PKR008
The truck PKR008 exists.

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.

7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 8
Truck rego: ABC001
Capacity: 1234.5
Weight: 234.5
Status: AVAILABLE

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 2
Rego: PKR768, Capacity: 1234.50, Weight: 3000.20, Status: AVAILABLE
Rego: SST005, Capacity: 12000.30, Weight: 50000.10, Status: USED
Rego: QRT834, Capacity: 5550.40, Weight: 400.50, Status: USED
Rego: LUCY01, Capacity: 3000.20, Weight: 1000.20, Status: AVAILABLE
Rego: KKK007, Capacity: 10000.50, Weight: 3000.40, Status: MAINTAINED
Rego: SYF777, Capacity: 3333.60, Weight: 4566.50, Status: MAINTAINED
Rego: PKR008, Capacity: 22000.30, Weight: 8800.50, Status: AVAILABLE
Rego: XCF003, Capacity: 30000.10, Weight: 10000.20, Status: AVAILABLE
Rego: GFT008, Capacity: 40000.30, Weight: 15000.40, Status: AVAILABLE
Rego: LUCY02, Capacity: 43000.20, Weight: 3000.50, Status: AVAILABLE
Rego: AL08UK, Capacity: 50000.10, Weight: 5000.00, Status: AVAILABLE
Rego: ABC001, Capacity: 1234.50, Weight: 234.50, Status: AVAILABLE

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 9
Trip number: 10
The trip 10 exists.

1. Display all employees.

2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 9
Trip number: 36
License: 123456
Rego: ABC001
Trip date (dd-mm-yyyy): 10-08-2023
Total legs: 2
Leg number: 1
Departure: Wollongong
Destination: Sydney
Leg number: 2
Departure: Sydney
Destination: Canberra

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 10
Data saved.

1. Display all employees.
2. Display all trucks.
3. Display all trips.
4. Find an employee.
5. Find a truck.
6. Find a trip.
7. Add a new employee.
8. Add a new truck.
9. Add a new trip.
10. Save all data into files.
0. Exit.
Input a choice (0-10): 0
Bye-bye

After the processing, the contents of updated files are listed below. New data are highlighted.

employees.txt
D, 1, John Smith, 02-12-1971, 42 Victoria St. Hurstville NSW 2456, 10001, AVAILABLE
D, 2, Peter Taylor, 01-12-1970, 42 Victoria St. Hurstville NSW 2456, 10008, ON LEAVE
D, 3, John Doe, 03-23-1966, 12 Station St. Dapto NSW 2530, 10002, AVAILABLE
D, 4, John Gray, 05-05-1988, 16 Station St. Dapto NSW 2530, 10004, AVAILABLE
D, 5, Adam Taylor, 01-01-1980, 42 Church St. City NSW 2300, 10003, ON LEAVE
D, 6, Michael Jones, 03-05-1975, 23 Waterloo Ave. Surry Hills NSW 2502, 10012, AVAILABLE
D, 7, Frederic Jones, 02-10-1978, 3 Victoria St. Redfern NSW 2420, 20002, BUSY
D, 8, Peter O'Brien, 02-28-1983, 19 Lucas Dr. Horsley NSW 2530, 20003, 'BUSY'
A, 9, John Lucas, 12-16-1966, 20 Huxley St. Horsley NSW 2530, SUPPORT
A, 10, John Fox, 10-15-1975, 18 Victoria St. Hurstville NSW 2456, DIRECTOR
D, 11, Adam Fox, 08-10-1974, 45 Victoria St. Hurstville NSW 2456, 30005, BUSY
D, 12, Phillip Cox, 12-12-1985, 5 The Avenue Rockdale NSW 2300, 40002, BUSY
D, 13, Andrew K Smith, 04-04-1969, 42 Bambaramba Ave. Pennant Hills NSW 2556, 20045, AVAILABLE
A, 14, Andrew R Smith, 04-01-1992, 67 King Cr. Hurstville NSW 2456, CEO
A, 15, Michael Potter, 04-01-1995, 568 Bong Bong St. Horsley NSW 2530, SUPPORT
D, 16, Harry Potter, 04-01-1987, 568 Bong Bong St. Horsley NSW 2530, 20055, AVAILABLE
D, 17, James Bond, 02-14-1989, 7 Alan Bond St. Perth WA 6000, 20065, AVAILABLE
D, 18, Paris Hilton, 05-01-1977, 1 Hilton St. Melbourne VIC 3000, 10305, AVAILABLE
A, 19, Lady Gaga, 06-01-1992, 3 Pork st. Hobart TAS 7000, SUPPORT
D, 20, Robin Hood, 05-23-1999, 6 Nottingham Pl. Sydney NSW 2000, 10345, AVAILABLE
D, 21, James Bond, 10-03-1992, 123 Princess Street Wollongong NSW 2500, 123456, AVAILABLE
A, 22, Albert Einstein, 08-02-1990, 20 Physicist road Wollongong NSW 2500, SUPPORT

trucks.txt
PKR768, 1234.50, 3000.20, AVAILABLE
SST005, 12000.30, 50000.10, USED
QRT834, 5550.40, 400.50, USED
LUCY01, 3000.20, 1000.20, AVAILABLE
KKK007, 10000.50, 3000.40, MAINTAINED
SYF777, 3333.60, 4566.50, MAINTAINED
PKR008, 22000.30, 8800.50, AVAILABLE
XCF003, 30000.10, 10000.20, AVAILABLE
GFT008, 40000.30, 15000.40, AVAILABLE
LUCY02, 43000.20, 3000.50, AVAILABLE
AL08UK, 50000.10, 5000.00, AVAILABLE
ABC001, 1234.50, 234.50, AVAILABLE

Trips.txt
1, 10001, PKR768, 01-12-2021
5
1, Sydney, Melbourne
2, Melbourne, Hobart
3, Hobart, Perth
4, Perth, Adelaide
5, Adelaide, Wollongong
2, 10001, SYF777, 02-20-2021
1
1, Sydney, Melbourne

3, 10001, KKK007, 03-12-2021
1
1, Sydney, Melbourne
4, 10001, PKR768, 06-29-2021
1
1, Sydney, Melbourne
5, 20002, PKR768, 01-12-2021
1
1, Melbourne, Sydney
6, 10002, SYF777, 02-20-2021
1
1, Melbourne, Sydney
7, 30005, KKK007, 03-12-2021
1
1, Melbourne, Sydney
8, 10001, PKR768, 01-13-2021
3
1, Sydney, Newcastle
2, Newcastle, Brisbane
3, Brisbane, Perth
9, 10002, QRT834, 09-17-2021
1
1, Sydney, Melbourne
10, 30005, KKK007, 12-15-2022
1
1, Sydney, Wollongong
11, 10003, SST005, 01-23-2022
1
1, Melbourne, Sydney
12, 10002, PKR768, 03-12-2022
1
1, Melbourne, Sydney
13, 20002, QRT834, 04-23-2022
2
1, Sydney, Melbourne
2, Melbourne, Sydney
14, 20002, PKR008, 04-23-2022
1
1, Wollongong, Sydney
15, 30005, PKR768, 05-24-2022
1
1, Sydney, Melbourne
16, 30005, SST005, 08-02-2022
1
1, Sydney, Wollongong
17, 20002, QRT834, 09-17-2022
2
1, Sydney, Wollongong
2, Wollongong, Sydney
18, 10001, KKK007, 12-15-2022
1
1, Melbourne, Sydney

19, 30005, SST005, 01-23-2022
1
1, Melbourne, Sydney
20, 10003, PKR768, 03-12-2022
1
1, Sydney, Melbourne
21, 10001, QRT834, 04-23-2022
1
1, Wollongong, Sydney
22, 30005, PKR008, 04-23-2022
1
1, Melbourne, Sydney
23, 10003, PKR768, 05-25-2022
1
1, Wollongong, Sydney
24, 20002, SST005, 08-02-2022
1
1, Sydney, Melbourne
25, 10001, PKR768, 01-12-2023
4
1, Melbourne, Sydney
2, Sydney, Perth
3, Perth, Sydney
4, Sydney, Brisbane
26, 10001, SYF777, 02-20-2023
1
1, Brisbane, Sydney
27, 20002, KKK007, 03-12-2023
1
1, Sydney, Wollongong
28, 30005, PKR768, 06-29-2023
2
1, Sydney, Melbourne
2, Melbourne, Perth
29, 10001, QRT834, 06-17-2023
1
1, Sydney, Perth
30, 10002, KKK007, 06-15-2023
2
1, Melbourne, Sydney
2, Sydney, Melbourne
31, 10003, SST005, 01-23-2023
1
1, Sydney, Melbourne
32, 20002, PKR768, 03-12-2023
1
1, Sydney, Melbourne
33, 30005, QRT834, 04-23-2023
1
1, Sydney, Melbourne
34, 30005, PKR008, 04-23-2023
1

1, Wollongong, Sydney
35, 10001, PKR768, 05-24-2023
4
1, Sydney, Melbourne
2, Melbourne, Adelaide
3, Adelaide, Perth
4, Perth, Sydney
36, 123456, ABC001, 10-08-2023
2
1, Wollongong, Sydney
2, Sydney, Canberra

Deliverables

(1) UML class diagram (1 mark): Use the UMLet application tool to draw the class diagram. The class diagram shall

- contains at least the classes mentioned above;
- contains the class name, fields, and methods definitions for each class;
- use correct and sufficient UML notations;
- specify the associations between classes;
- specify the multiplicities for both sides of the associations.

Remember to use the `CSIT213` palette!

Use the option File->Export as… to export a class diagram into a file in `BMP` format. Do not delete an exported file. You will use it as one of the solutions for your task.

Insert the BMP files into a Word file `assignment2Report.docx`.

(2) Implementation (12 marks): Implement the application according to the UML class diagrams and the UML activity diagrams described above. The program shall

- be consistent with the UML class diagrams.
- follow the conventions for naming all classes, variables, and methods.
- provide sufficient comments.
- use proper blank spaces, indentation, and braces to make your code easy to read and understand.
- follow the specified implementation steps.
- be able to repeat the main menu until the user exits the system.

(3) Compilation and test (2 marks): Compilation and test your Java program by using the command line interface.

- Please carefully compile your program. Make sure your program can pass the compilation by using the `javac` command.
- Test your program by using the `java` command.
  Test your program for all the activities. See the examples of the processing results above for more details.
- **Please do not define the package in your program (a special alert for students who use IDE to complete the assignment).**

Copy and paste the compilation and testing results into the Word file `assignment2Report.docx`. When ready convert the Word file `assignment2Report.docx` into a pdf file `assignment2Report.pdf`.

Submission

Note that you have only one submission. So, make absolutely sure that you submit the correct files with the correct contents and correct types. No other submission is possible!

Submit the files **`TripManagement.java`**, **`Employee.java`**, **`Truck.java`**, **`Trip.java`**, **`MyFileIO.java`** and **`assignment2Report.pdf`** through Moodle in the following way:

(1) Access Moodle at **`http://moodle.uowplatform.edu.au/`**
(2) To login use a **`Login`** link located in the right upper corner of the Web page or in the middle of the bottom of the Web page
(3) When logged select a site **`CSIT213 (S223) Java Programming`**
(4) Scroll down to a section **`Assignments and Submissions`**
(5) Click on a link **`Assignment 2 submission`**
(6) Click on the button **`Add Submission`**
(7) Upload a file **`TripManagement.java`** into an area **`You can drag and drop files here to add them`**. You can also use a link **`Add`**...
(8) Repeat step (7) for the files **`Employee.java`**, **`Truck.java`**, **`Trip.java`**, **`MyFileIO.java`** and **`assignment2Report.pdf`**.
(9) Click on the checkbox with a text attached: **`By checking this box, I confirm that this submission is my own work,`** ... in order to confirm the authorship of your submission
(10) Click on a button **`Save changes`**

End of specification