School of Computing and Information Technology
University of Wollongong

Session: **Autumn 2023**
Lecturer: **Tianbing Xia**

CSIT1121 Object Oriented Design and Programming
Assignment 2

Published on 27 March 2023

Scope

This assignment is related to the class conceptual schema, Java classes' definition and implementation, interface, abstract, polymorphism, and file I/O.

Please read very carefully the information listed below.

This assignment contributes to 10% of the total evaluation in the subject CSIT121.

The outcomes of the assignment work are due by **Saturday 29 April 2023, 11.55 pm (sharp).**

General Java coding requirements:

- You should 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.
- You should create identifiers with sensible names.
- You should make 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: CSIT121
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

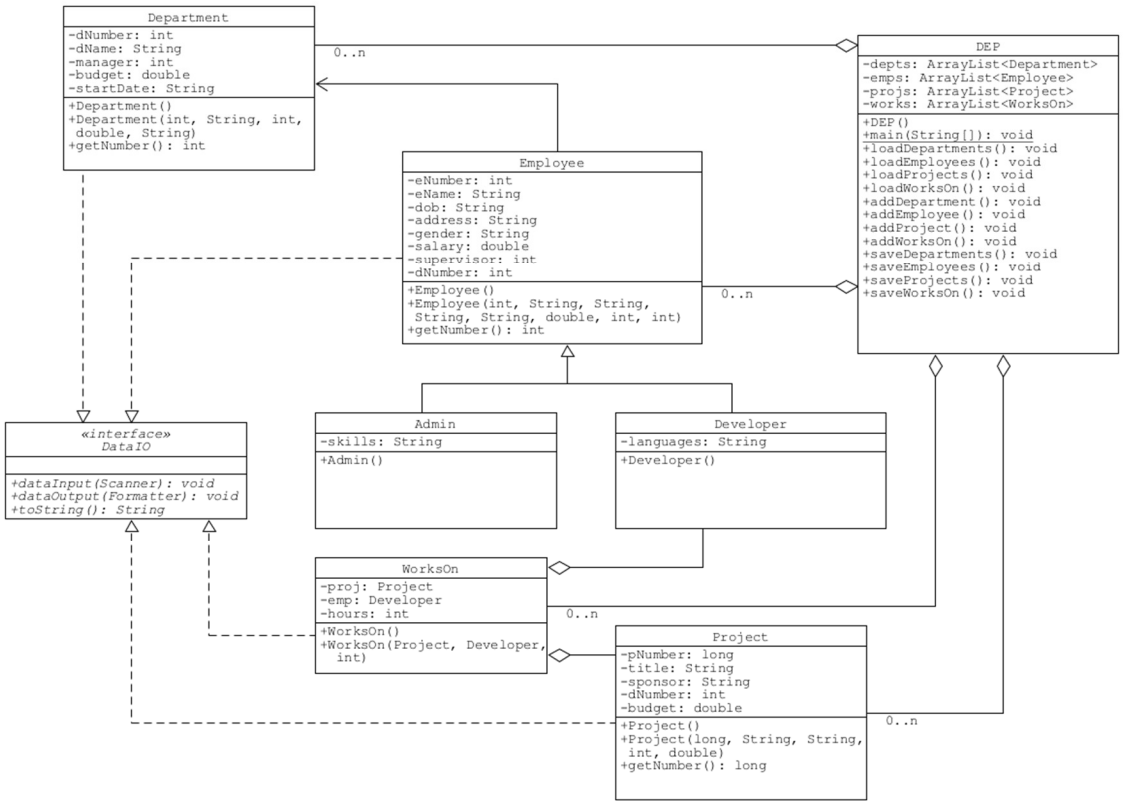
In this assignment, you are required to design and implement a Department Employee and Project (DEP) System in Java. This system helps a company to manage employees and projects. The related topics include interface, abstract, polymorphism, and file I/O.

Implementation

The DEP system helps a company to manage departments, employees, and projects. The system load data of departments, employees, projects, and works-on 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 departments, employees, projects, or works-on. A user can add a new department, a new employee, a new project, or new work-on information. A user can save the updated data into text files.

The UML class diagram of DEP 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 `DataIO` in a file DataIO.java that only consists of three abstract methods: `dataInput(Scanner)`, `dataOutput(Formatter)` and `toString()`.

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

Define a class `Department` in a file `Department.java` that implements the interface `DataIO`.

Define a class `Project` in a file `Project.java` that implements the interface `DataIO`.

Define a class `WorksOn` in a file `WorksOn.java` that implements the interface `DataIO`.

The `DEP` class is defined in a file `DEP.java` that contains `ArrayList` containers containing objects of `Department`, `Employee`, `Project` and `WorksOn`.

The DEP application initially loads data of departments, employees, projects, and works-on from text files `departments.txt`, `employees.txt`, `projects.txt` and `workson.txt` and stores them in the containers by calling the methods loadDepartments(), loadEmployees(), loadProjects() and loadWorksOn().

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

1, SALES, 110, 1234.00, 02/01/2012
2, ACCOUNTING, 120, 5566789.50, 30/10/2010
3, GAMES, 150, 100000.00, 01/03/2008

Each row is a record of a department. It contains a department number, name, manager number, budgets and manager start date. Each field is separated by a comma(,) and a space ( ).

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

A, 600, Willy, 01/01/1988, 41 Station Street Wollongong NSW 2500, M, 250.5, 0, 9, Cook Read
D, 700, Zhi, 12/09/1999, 112 Smith Street Windang NSW 2525, M, 80.2, 600, 9, C++ Java Python
D, 800, Mary, 03/10/2000, 26 Gibsons Road Figtree NSW 2525, F, 50.0, 700, 9, Java Python SQL

Each row is a record of an employee. The first character is the type of employee. The letter `D` means it is a Developer 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, gender, salary, supervisor number, department number, and skills.
A developer record contains an employee number, name, date of birth, address, gender, salary, supervisor number, department number, and programming languages.

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

1001, Computation, Microsoft, 8, 25000
1002, Study methods, Education committee, 3, 15000

Each row is a record of a project. A project record contains a project number, title, sponsor, department number and budget. Each field is separated by a comma (,) and a space ( ).

The format of a file workson.txt contains data like

1001, Computation, Microsoft, 8, 25000
800, Mary, 03/10/2000, 26 Gibsons Road Figtree NSW 2525, F, 50.0, 700, 9, Java Python SQL
10

A works-on record consists of a project record, a developer record and the hours in three rows.

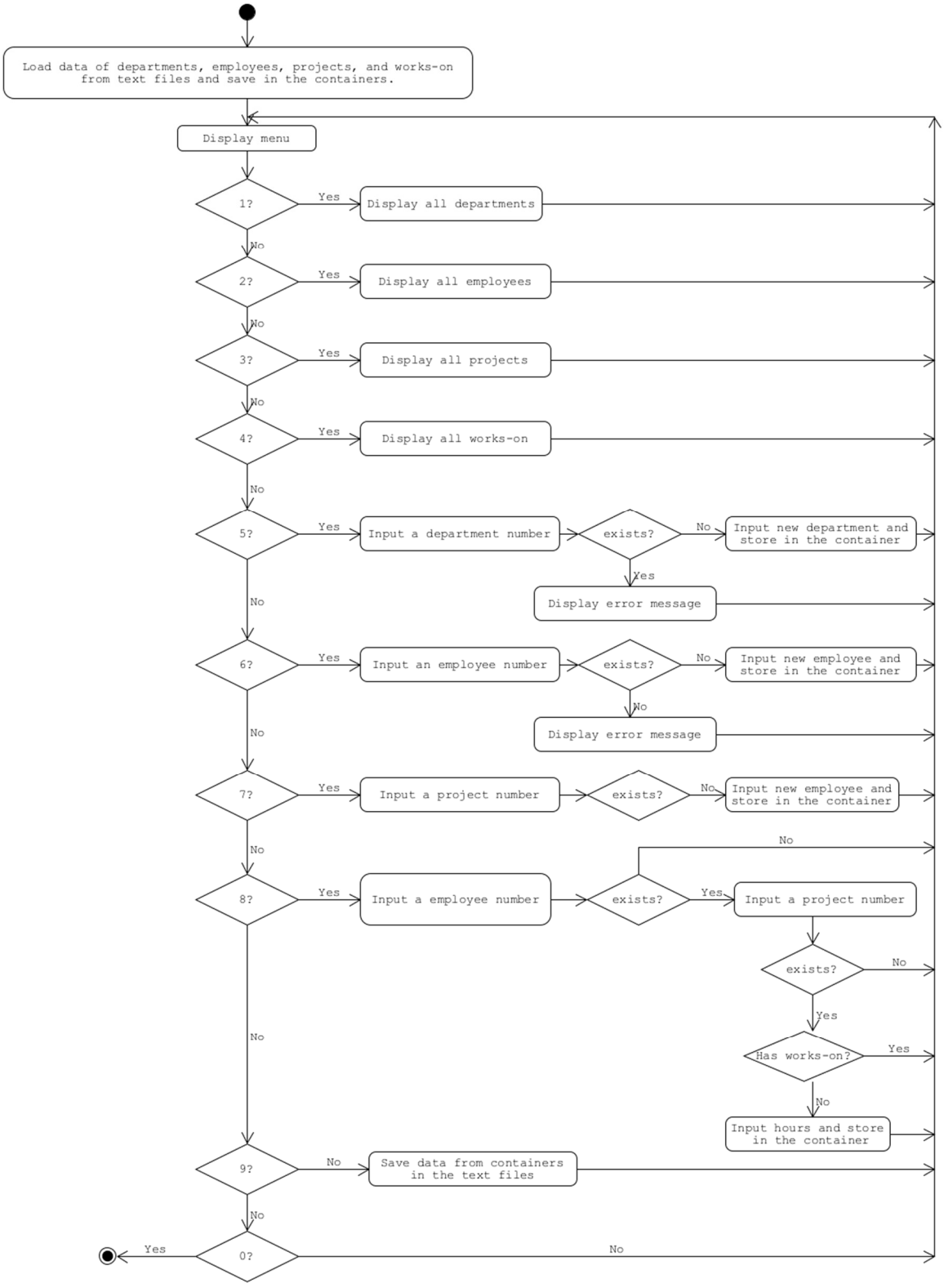
Hint: You can open a text file, and then call the method
`useDelimiter(", |\r\n|\n")`
of a `Scanner` object before getting input data from a text file.

After loading the data into the containers, the main() method display 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 departments' information from the container `depts`.
- If an input value is 2, the program calls a method to display all employees' information from the container `emps`.
- If an input value is 3, the program calls a method to display all projects' information from the container `projs`.
- If an input value is 4, the program calls a method to display all works-on information from the container `works`.
- If an input value is 5, the program calls a method `addDepartment()`. A user inputs a department number. The method searches the department container. If the department does not exist, input the new department, and store it in the container.
- If an input value is 6, the program calls a method `addEmployee()`. A user inputs an employee number. The method searches the employee container. If the employee does not exist, input the new employee, and store it in the container.
- If an input value is 7, the program calls a method `addProject()`. A user inputs a project number. The method searches the project container. If the project does not exist, input the new project, and store it in the container.

- If an input value is 8, the program calls a method `addWorksOn()`. A user inputs an employee number. If the employee exists, the user inputs a project number. If the project exists and the employee does not work on the project, the user inputs the total hours and stores the works-on object in the container.
- If an input value is 9, the program calls methods `saveDepartments()`, `saveEmployees()`, `saveProjects()` and `saveWorksOn()` to save the data from containers in the text files. The format of each text file must be the same as the test files `departments.txt`, `employees.txt`, `projects.txt`, and `workson.txt` displayed above.
- If an input value is 0, the program displays the message `Bye-bye`, and then exits.

UML activity diagram



Testing files

Download files `departments.txt`, `employees.txt`, `projects.txt` and `workson.txt` from the Moodle site.

Compilation and testing

Compile your program by using the `javac` command.

```
javac DEP.java
```

Process your program by using the `java` command.

```
java DEP
```

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 departments.
2. Display all employees.
3. Display all projects.
4. Display information for employees who works on projects.
5. Add a new department.
6. Add a new employee.
7. Add a new project.
8. Add a new works on record.
9. Save all data into files.
0. Exit.
Input a choice (0-9): 1
Department number: 1, Department name: SALES, Manager number:
   110, Budget: 1234.0, Manager start date: 02/01/2012
Department number: 2, Department name: ACCOUNTING, Manager
   number: 120, Budget: 5566789.5, Manager start date:
   30/10/2010
Department number: 3, Department name: GAMES, Manager number:
   150, Budget: 100000.0, Manager start date: 01/03/2008
Department number: 4, Department name: HUMAN RESOURCES,
   Manager number: 200, Budget: 500000.0, Manager start date:
   02/01/2013
Department number: 5, Department name: SPORTS, Manager
   number: 250, Budget: 8500000.0, Manager start date:
   10/05/2010
```

```
Department number: 6, Department name: RESEARCH, Manager
   number: 300, Budget: 45500.0, Manager start date:
   10/06/2020
Department number: 7, Department name: EDUCATION, Manager
   number: 350, Budget: 100000.0, Manager start date:
   10/07/2019
Department number: 8, Department name: FINANCE, Manager
   number: 500, Budget: 8400000.0, Manager start date:
   10/08/2022
Department number: 9, Department name: COMPUTING, Manager
   number: 600, Budget: 90000.0, Manager start date:
   10/09/2018

1. Display all departments.
2. Display all employees.
3. Display all projects.
4. Display information for employees who works on projects.
5. Add a new department.
6. Add a new employee.
7. Add a new project.
8. Add a new works on record.
9. Save all data into files.
0. Exit.
Input a choice (0-9): 2
Admin Employee number: 600, Employee name: Willy, Date of
   birth: 01/01/1988, Address: 41 Station Street Wollongong
   NSW 2500, Gender: M, Salary: 250.5, Supervisor number: 0,
   Department number: 9, Skills: Cook Read
Developer Employee number: 700, Employee name: Zhi, Date of
   birth: 12/09/1999, Address: 112 Smith Street Windang NSW
   2525, Gender: M, Salary: 80.2, Supervisor number: 600,
   Department number: 9, Languages: C++ Java Python
Developer Employee number: 800, Employee name: Mary, Date of
   birth: 03/10/2000, Address: 26 Gibsons Road Figtree NSW
   2525, Gender: F, Salary: 50.0, Supervisor number: 700,
   Department number: 9, Languages: Java Python SQL
Admin Employee number: 500, Employee name: Angelina, Date of
   birth: 20/11/1990, Address: 25 Bong Bong Road Horsley NSW
   2530, Gender: F, Salary: 250.0, Supervisor number: 0,
   Department number: 8, Skills: Read Write
Developer Employee number: 510, Employee name: Anna, Date of
   birth: 20/11/1990, Address: 200 Cemetary Road Hawea NSW
   2800, Gender: F, Salary: 100.0, Supervisor number: 500,
   Department number: 8, Languages: Python SQL
Admin Employee number: 520, Employee name: Madelaine, Date of
   birth: 20/11/1990, Address: 23 Lake View Street Figtree
```

NSW 2525, Gender: F, Salary: 50.0, Supervisor number: 510, Department number: 8, Skills: Write Communication
Admin Employee number: 530, Employee name: Robert, Date of birth: 20/11/1990, Address: 80 Penny Road Windang NSW 2520, Gender: M, Salary: 50.0, Supervisor number: 510, Department number: 8, Skills: Management Travel
Developer Employee number: 540, Employee name: Claudio, Date of birth: 20/11/1990, Address: 23 Horsley Street Unanderra NSW 2528, Gender: M, Salary: 50.0, Supervisor number: 510, Department number: 8, Languages: Java C Python SQL
Developer Employee number: 350, Employee name: Brian, Date of birth: 13/05/1965, Address: 23 Station Street Wollongong NSW 2500, Gender: M, Salary: 200.4, Supervisor number: 0, Department number: 7, Languages: Shell Java Python
Developer Employee number: 110, Employee name: Alvin, Date of birth: 13/10/1977, Address: 56 Marlo Road Wollongong NSW 2500, Gender: M, Salary: 156.4, Supervisor number: 100, Department number: 1, Languages: SQL Shell HTML

1. Display all departments.
2. Display all employees.
3. Display all projects.
4. Display information for employees who works on projects.
5. Add a new department.
6. Add a new employee.
7. Add a new project.
8. Add a new works on record.
9. Save all data into files.
0. Exit.

Input a choice (0-9): 3
Project number: 1001, Project title: Computation, Sponsor: Microsoft, Department number: 8, Project budget: 25000.0
Project number: 1002, Project title: Study methods, Sponsor: Education committee, Department number: 3, Project budget: 15000.0
Project number: 1003, Project title: Racing car, Sponsor: Cloud Pty Ltd, Department number: 3, Project budget: 225000.0
Project number: 1004, Project title: Football, Sponsor: Football Club, Department number: 5, Project budget: 35000.0
Project number: 1005, Project title: Swimming, Sponsor: Education Committee, Department number: 5, Project budget: 125000.0
Project number: 1006, Project title: Database, Sponsor: Database Committee, Department number: 5, Project budget: 125000.0

```
1. Display all departments.
2. Display all employees.
3. Display all projects.
4. Display information for employees who works on projects.
5. Add a new department.
6. Add a new employee.
7. Add a new project.
8. Add a new works on record.
9. Save all data into files.
0. Exit.
Input a choice (0-9): 4
Project number: 1001, Project title: Computation, Sponsor:
   Microsoft, Department number: 8, Project budget: 25000.0
Developer Employee number: 800, Employee name: Mary, Date of
   birth: 03/10/2000, Address: 26 Gibsons Road Figtree NSW
   2525, Gender: F, Salary: 50.0, Supervisor number: 700,
   Department number: 9, Languages: Java Python SQL
Hours: 10

Project number: 1001, Project title: Computation, Sponsor:
   Microsoft, Department number: 8, Project budget: 25000.0
Developer Employee number: 510, Employee name: Anna, Date of
   birth: 20/11/1990, Address: 200 Cemetary Road Hawea NSW
   2800, Gender: F, Salary: 100.0, Supervisor number: 500,
   Department number: 8, Languages: Python SQL
Hours: 15

Project number: 1002, Project title: Study methods, Sponsor:
   Education committee, Department number: 3, Project budget:
   15000.0
Developer Employee number: 700, Employee name: Zhi, Date of
   birth: 12/09/1999, Address: 112 Smith Street Windang NSW
   2525, Gender: M, Salary: 80.2, Supervisor number: 600,
   Department number: 9, Languages: C++ Java Python
Hours: 20

Project number: 1003, Project title: Racing car, Sponsor:
   Cloud Pty Ltd, Department number: 3, Project budget:
   225000.0
Developer Employee number: 800, Employee name: Mary, Date of
   birth: 03/10/2000, Address: 26 Gibsons Road Figtree NSW
   2525, Gender: F, Salary: 50.0, Supervisor number: 700,
   Department number: 9, Languages: Java Python SQL
Hours: 15
```

```
Project number: 1003, Project title: Racing car, Sponsor:
   Cloud Pty Ltd, Department number: 3, Project budget:
   225000.0
Developer Employee number: 540, Employee name: Claudio, Date
   of birth: 20/11/1990, Address: 23 Horsley Street Unanderra
   NSW 2528, Gender: M, Salary: 50.0, Supervisor number: 510,
   Department number: 8, Languages: Java C Python SQL
Hours: 10

Project number: 1004, Project title: Football, Sponsor:
   Football Club, Department number: 5, Project budget:
   35000.0
Developer Employee number: 350, Employee name: Brian, Date of
   birth: 13/05/1965, Address: 23 Station Street Wollongong
   NSW 2500, Gender: M, Salary: 200.4, Supervisor number: 0,
   Department number: 7, Languages: Shell Java Python
Hours: 10

Project number: 1005, Project title: Swimming, Sponsor:
   Education Committee, Department number: 5, Project budget:
   125000.0
Developer Employee number: 350, Employee name: Brian, Date of
   birth: 13/05/1965, Address: 23 Station Street Wollongong
   NSW 2500, Gender: M, Salary: 200.4, Supervisor number: 0,
   Department number: 7, Languages: Shell Java Python
Hours: 10


1. Display all departments.
2. Display all employees.
3. Display all projects.
4. Display information for employees who works on projects.
5. Add a new department.
6. Add a new employee.
7. Add a new project.
8. Add a new works on record.
9. Save all data into files.
0. Exit.
Input a choice (0-9): 5
Department number: 3
The department 3 exists.

1. Display all departments.
2. Display all employees.
3. Display all projects.
4. Display information for employees who works on projects.
5. Add a new department.
```

```
6. Add a new employee.
7. Add a new project.
8. Add a new works on record.
9. Save all data into files.
0. Exit.
Input a choice (0-9): 5
Department number: 10
Department name: Database system
Manager number: 520
Budget: 15000.5
Start date: 10/01/2020

1. Display all departments.
2. Display all employees.
3. Display all projects.
4. Display information for employees who works on projects.
5. Add a new department.
6. Add a new employee.
7. Add a new project.
8. Add a new works on record.
9. Save all data into files.
0. Exit.
Input a choice (0-9): 6
Employee number: 110
The employee 110 exists.

1. Display all departments.
2. Display all employees.
3. Display all projects.
4. Display information for employees who works on projects.
5. Add a new department.
6. Add a new employee.
7. Add a new project.
8. Add a new works on record.
9. Save all data into files.
0. Exit.
Input a choice (0-9): 6
Employee number: 120
Employee name: Alice Smith
Data of birth (dd/mm/yyyy): 10/02/2000
Address: 1 Northfields ave Wollongong NSW 2500
Gender: F
Salary: 520.3
Supervisor number: 110
Department number: 3
Admin or Developer (A or D)?: D
Languages: C++ Java SQL Lisp
```

```
1. Display all departments.
2. Display all employees.
3. Display all projects.
4. Display information for employees who works on projects.
5. Add a new department.
6. Add a new employee.
7. Add a new project.
8. Add a new works on record.
9. Save all data into files.
0. Exit.
Input a choice (0-9): 7
Project number: 1003
The project 1003 exists.

1. Display all departments.
2. Display all employees.
3. Display all projects.
4. Display information for employees who works on projects.
5. Add a new department.
6. Add a new employee.
7. Add a new project.
8. Add a new works on record.
9. Save all data into files.
0. Exit.
Input a choice (0-9):7
Project number: 1008
Project title: Database Management
Sponsor: Oracle Pty Ltd
Department number: 9
Budget: 100000.0

1. Display all departments.
2. Display all employees.
3. Display all projects.
4. Display information for employees who works on projects.
5. Add a new department.
6. Add a new employee.
7. Add a new project.
8. Add a new works on record.
9. Save all data into files.
0. Exit.
Input a choice (0-9): 8
Employee number: 100
The employee 100 does not exist.

1. Display all departments.
```

```
2. Display all employees.
3. Display all projects.
4. Display information for employees who works on projects.
5. Add a new department.
6. Add a new employee.
7. Add a new project.
8. Add a new works on record.
9. Save all data into files.
0. Exit.
Input a choice (0-9): 8
Employee number: 120
Project number: 1004
Input total hours: 20

1. Display all departments.
2. Display all employees.
3. Display all projects.
4. Display information for employees who works on projects.
5. Add a new department.
6. Add a new employee.
7. Add a new project.
8. Add a new works on record.
9. Save all data into files.
0. Exit.
Input a choice (0-9): 8
Employee number: 540
Project number: 1003
The employee 540 has already been allocated to the project
   1003

1. Display all departments.
2. Display all employees.
3. Display all projects.
4. Display information for employees who works on projects.
5. Add a new department.
6. Add a new employee.
7. Add a new project.
8. Add a new works on record.
9. Save all data into files.
0. Exit.
Input a choice (0-9): 9
Data saved.

1. Display all departments.
2. Display all employees.
3. Display all projects.
4. Display information for employees who works on projects.
```

```
5. Add a new department.
6. Add a new employee.
7. Add a new project.
8. Add a new works on record.
9. Save all data into files.
0. Exit.
Input a choice (0-9): 0
Bye-bye
```

The contents of updated files are listed below. New data are highlighted.

departments.txt:

```
1, SALES, 110, 1234.00, 02/01/2012
2, ACCOUNTING, 120, 5566789.50, 30/10/2010
3, GAMES, 150, 100000.00, 01/03/2008
4, HUMAN RESOURCES, 200, 500000.00, 02/01/2013
5, SPORTS, 250, 8500000.00, 10/05/2010
6, RESEARCH, 300, 45500.00, 10/06/2020
7, EDUCATION, 350, 100000.00, 10/07/2019
8, FINANCE, 500, 8400000.00, 10/08/2022
9, COMPUTING, 600, 90000.00, 10/09/2018
10, Database system, 520, 15000.50, 10/01/2020
```

employees.txt:

```
A, 600, Willy, 01/01/1988, 41 Station Street Wollongong NSW 2500, M, 250.50, 0, 9, Cook Read
D, 700, Zhi, 12/09/1999, 112 Smith Street Windang NSW 2525, M, 80.20, 600, 9, C++ Java Python
D, 800, Mary, 03/10/2000, 26 Gibsons Road Figtree NSW 2525, F, 50.00, 700, 9, Java Python SQL
A, 500, Angelina, 20/11/1990, 25 Bong Bong Road Horsley NSW 2530, F, 250.00, 0, 8, Read Write
D, 510, Anna, 20/11/1990, 200 Cemetary Road Hawea NSW 2800, F, 100.00, 500, 8, Python SQL
A, 520, Madelaine, 20/11/1990, 23 Lake View Street Figtree NSW 2525, F, 50.00, 510, 8, Write Communication
A, 530, Robert, 20/11/1990, 80 Penny Road Windang NSW 2520, M, 50.00, 510, 8, Management Travel
D, 540, Claudio, 20/11/1990, 23 Horsley Street Unanderra NSW 2528, M, 50.00, 510, 8, Java C Python SQL
D, 350, Brian, 13/05/1965, 23 Station Street Wollongong NSW 2500, M, 200.40, 0, 7, Shell Java Python
D, 110, Alvin, 13/10/1977, 56 Marlo Road Wollongong NSW 2500, M, 156.40, 100, 1, SQL Shell HTML
D, 120, Alice Smith, 10/02/2000, 1 Northfields ave Wollongong NSW 2500, F, 520.30, 110, 3, C++ Java SQL Lisp
```

projects.txt:
1001, Computation, Microsoft, 8, 25000.00
1002, Study methods, Education committee, 3, 15000.00
1003, Racing car, Cloud Pty Ltd, 3, 225000.00
1004, Football, Football Club, 5, 35000.00
1005, Swimming, Education Committee, 5, 125000.00
1006, Database, Database Committee, 5, 125000.00
1008, Database Management, Oracle Pty Ltd, 9, 100000.00

workson.txt:
1001, Computation, Microsoft, 8, 25000.00
800, Mary, 03/10/2000, 26 Gibsons Road Figtree NSW 2525, F, 50.00, 700, 9, Java Python SQL
10
1001, Computation, Microsoft, 8, 25000.00
510, Anna, 20/11/1990, 200 Cemetary Road Hawea NSW 2800, F, 100.00, 500, 8, Python SQL
15
1002, Study methods, Education committee, 3, 15000.00
700, Zhi, 12/09/1999, 112 Smith Street Windang NSW 2525, M, 80.20, 600, 9, C++ Java Python
20
1003, Racing car, Cloud Pty Ltd, 3, 225000.00
800, Mary, 03/10/2000, 26 Gibsons Road Figtree NSW 2525, F, 50.00, 700, 9, Java Python SQL
15
1003, Racing car, Cloud Pty Ltd, 3, 225000.00
540, Claudio, 20/11/1990, 23 Horsley Street Unanderra NSW 2528, M, 50.00, 510, 8, Java C
Python SQL
10
1004, Football, Football Club, 5, 35000.00
350, Brian, 13/05/1965, 23 Station Street Wollongong NSW 2500, M, 200.40, 0, 7, Shell Java
Python
10
1005, Swimming, Education Committee, 5, 125000.00
350, Brian, 13/05/1965, 23 Station Street Wollongong NSW 2500, M, 200.40, 0, 7, Shell Java
Python
10
1004, Football, Football Club, 5, 35000.00
120, Alice Smith, 10/02/2000, 1 Northfields ave Wollongong NSW 2500, F, 520.30, 110, 3, C++
Java SQL Lisp
20
1006, Database, Database Committee, 5, 125000.00
540, Claudio, 20/11/1990, 23 Horsley Street Unanderra NSW 2528, M, 50.00, 510, 8, Java C
Python SQL
15

Deliverables

(1) UML class diagram (1 mark): Use the UMLet application tool to complete/update the program design based on the given UML class diagram. You CANNOT modify or delete any existing class, interface, attributes, and methods. You CAN add new classes, associations, and methods.

Remember to use the CSIT121 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, e.g. DEP.docx.

(2) Implementation (7 marks): Implement the DEP system according to the UML class diagrams and implementation descriptions above. The program shall

- assume the customer's inputs are always correct and valid.
- 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.
- be implemented by using the advanced OOP features such as inheritance, polymorphism, abstraction, interface, and `instanceof` class.

(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 items. 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 DEP.docx.
When ready convert the Word file DEP.docx into a pdf file DEP.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 **DEP.java**, **DataIO.java, Department.java, Employee.java, Project.java, WorksOn.java and DEP.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 **CSIT121 (S123) Object Oriented Design and 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) Move a file **DEP.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 **DataIO.java, Department.java, Employee.java, Project.java, WorksOn.java and DEP.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 the button **Save changes**

End of specification