

```
1
2 package dao;
3
4 import adt.*;
5
6 import entity.Course;
7 import entity.CourseProgram;
8 import entity.Program;
9 import entity.Student;
10 import entity.Tutor;
11 import entity.Tutorial;
12 import entity.TutorialGroup;
13 import entity.TutorialProgram;
14 import java.io.IOException;
15
16 /**
17  *
18  * @author Chew Lip Sin
19  * @author Eugene Teoh
20  * @author Fung Chun Xiang
21  * @author Lim Yi Leong
22  */
23 public class Initializer {
24
25     // Method to return a collection of with hard-coded entity values
26     public ListInterface<Course> initializeCourse() {
27         ListInterface<Course> cList = new ArrList<>();
28         cList.add(new Course("BAIT1013", "Introduction of Computer Networks", 4,
Course.Sem.JAN));
29         cList.add(new Course("BAMS1623", "Discrete Mathematics", 3, Course.Sem.ALL));
30         cList.add(new Course("BAMS1054", "Probability and Statistics", 4,
Course.Sem.JUL));
31         cList.add(new Course("BJEL1713", "English for Tertiary Studies", 3,
Course.Sem.JAN));
32         cList.add(new Course("BJEL1723", "Academic English", 3, Course.Sem.JUL));
33         cList.add(new Course("BACS1053", "Database Management", 4, Course.Sem.ALL));
34         cList.add(new Course("BMCS2013", "Data Engineering", 4, Course.Sem.ALL));
35         cList.add(new Course("AACS1304", "System Analysis and Design", 3,
Course.Sem.JUL));
36         cList.add(new Course("BAIT1043", "System Analysis and Design", 3,
Course.Sem.JAN));
37         cList.add(new Course("AMIT1713", "IT Fundamentals and Applications", 3,
Course.Sem.ALL));
38         cList.add(new Course("BMIT1723", "IT Fundamentals and Applications", 3,
Course.Sem.ALL));
39         cList.add(new Course("ABFA1173", "Principles of Accounting ", 3,
```

```

Course.Sem.ALL));
40      cList.add(new Course("BBFA1173", "Principles of Accounting ", 3,
Course.Sem.ALL));
41      cList.add(new Course("BBMF3203", "Research Method", 3, Course.Sem.ALL));
42      cList.add(new Course("MPU-2312", "Civic Consciousness and Volunteerism", 2,
Course.Sem.ALL));
43      cList.add(new Course("MPU-2212", "Bahasa Kebangsaan A", 2, Course.Sem.ALL));
44
45      return cList;
46  }
47
48  public LinkedListInterface<CourseProgram> CourseProgramInitializer() {
49      LinkedListInterface<CourseProgram> cP = new DoublyLinkedList<>();
50      cP.add(new CourseProgram("BMIT1713", "RSD", false));
51      cP.add(new CourseProgram("BBMF3203", "RSD", false));
52      cP.add(new CourseProgram("BBMF3203", "RME", false));
53      cP.add(new CourseProgram("BAIT1013", "RSD", false));
54
55      return cP;
56  }
57
58  public ListInterface<Tutor> initializeTutors() {
59
60      ListInterface<Tutor> ttList = new ArrList<>();
61      ttList.add(new Tutor("Ng Yen Phing", "ngyp@tarc.edu.my", 1, "Management
Information Systems"));
62      ttList.add(new Tutor("Ho Chuk Fong", "hocf@tarc.edu.my", 2, "Accounting &
Finance"));
63      ttList.add(new Tutor("Chin Wan Yoke", "chinwy@tarc.edu.my", 2, "Banking &
Finance"));
64      ttList.add(new Tutor("Ong Ting Hao", "ongth@tarc.edu.my", 5, "Exercise
Physiology"));
65      ttList.add(new Tutor("Goh Ching Pang", "gohcp@tarc.edu.my", 3, "Artificial
Intelligence"));
66      ttList.add(new Tutor("Lee Fong Yee", "leefy@tarc.edu.my", 4, "Bioscience"));
67      return ttList;
68  }
69
70  public static ListInterface<TutorialGroup> initializeTutorialGroups() {
71      ListInterface<TutorialGroup> tutorialGroups = new ArrList<>();
72
73      tutorialGroups.add(new TutorialGroup("RDS2S1G1"));
74      tutorialGroups.add(new TutorialGroup("RAC2S1G1"));
75      tutorialGroups.add(new TutorialGroup("RDS2S1G2"));
76      tutorialGroups.add(new TutorialGroup("RDS2S1G3"));
77      tutorialGroups.add(new TutorialGroup("RAC2S1G2"));

```

```

78         tutorialGroups.add(new TutorialGroup("RSW2S1G1"));
79
80         tutorialGroups.getEntry(1).addStudent(new Student("Fung Chun Xiang",
"23WMMR09149", "funcx-pm21@student.tarc.edu.my"));
81         tutorialGroups.getEntry(1).addStudent(new Student("Eugene Teoh Kai Siang",
"23WMMR09147", "eugenetks-pm21@student.tarc.edu.my"));
82         tutorialGroups.getEntry(2).addStudent(new Student("Tee Wen Xin", "23WMMR08480",
"teewx-wm21@student.tarc.edu.my"));
83         tutorialGroups.getEntry(3).addStudent(new Student("Khor Jinn Zhi", "23WMMR09176",
"khorjz-pm21@student.tarc.edu.my"));
84         tutorialGroups.getEntry(3).addStudent(new Student("Lee Tze Siaen", "23WMMR09179",
"leets-pm21@student.tarc.edu.my"));
85
86         return tutorialGroups;
87     }
88
89     // Method to return a collection of with hard-coded entity values
90     public ListInterface<Tutorial> TutorialInitializer() {
91         ListInterface<Tutorial> tList = new ArrList<>();
92         tList.add(new Tutorial("RSDG1"));
93         tList.add(new Tutorial("RBAG1"));
94         tList.add(new Tutorial("RACG1"));
95         tList.add(new Tutorial("RBFG1"));
96         tList.add(new Tutorial("RMEG1"));
97         tList.add(new Tutorial("RSDG2"));
98         tList.add(new Tutorial("RBAG2"));
99         tList.add(new Tutorial("RACG2"));
100         tList.add(new Tutorial("RBFG2"));
101         tList.add(new Tutorial("RMEG2"));
102
103         return tList;
104     }
105
106     public static void inputTutorial() {
107         Initializer t = new Initializer();
108         ListInterface<Tutorial> tList = t.TutorialInitializer();
109
110         // Save the program data to the file using ProgramDAO
111         DAO DAO = new DAO();
112         DAO.saveToFile(tList, "tutorial.dat");
113
114         System.out.println("\nProgram: \n" + tList);
115     }
116
117     // Method to return a collection of with hard-coded entity values
118     public ListInterface<Program> ProgramInitializer() {

```

```
119         ListInterface<Program> pList = new ArrList<>();
120         pList.add(new Program("RSD", "Bachelor Degree", "Bachelor of Information
Technology (Honours) in Software Systems Development", "FOCS - Faculty of Computing and
Information Technology", "This programme produces and equips graduates with in-depth
knowledge and skills that are essential to work as professionals in the software systems
development and computer networking sectors."));
121         pList.add(new Program("RBA", "Bachelor Degree", "Bachelor of Business
Administration (Honours)", "FAFB - Faculty of Accountancy, Finance & Business", "This
programme provides students with a well-rounded business education through a core curriculum
of management, marketing, finance, accounting, economics and law courses."));
122         pList.add(new Program("RAC", "Bachelor Degree", "Bachelor of Accounting
(Honours)", "FAFB - Faculty of Accountancy, Finance & Business", "This programme provides a
wide spectrum of knowledge and skills required for a career in the accountancy and finance
profession."));
123         pList.add(new Program("RBF", "Bachelor Degree", "Bachelor of Banking and Finance
(Honours)", "FAFB - Faculty of Accountancy, Finance & Business", "This programme aims to
produce graduates who are equipped with the knowledge and core competencies and dispositions
necessary for pursuing a demanding career in the highly competitive domestic and
international banking and finance industry."));
124         pList.add(new Program("RME", "Bachelor Degree", "Bachelor of Mechanical
Engineering with Honours", "FOET - Faculty of Engineering and Technology", "Mechanical
engineering is the broadest among all engineering disciplines. Thus, most of the modern-day
inventions are due to knowledge and application of mechanical engineering."));
125         pList.add(new Program("DAC", "Diploma", "Diploma in Accounting", "FAFB - Faculty
of Accountancy, Finance & Business", "This programme provides students with broad base
knowledge in financial accounting & reporting, management accounting, finance, economics,
management, taxation, auditing, law and information technology & systems."));
126         pList.add(new Program("DMC", "Diploma", "Diploma in Mass Communication
(Advertising)", "FCCI - Faculty of Communication & Creative Industries", "This programme
encompasses the basic knowledge and practical skills of advertising, media and
communication."));
127         pList.add(new Program("DCS", "Diploma", "Diploma in Computer Science", "FOCS -
Faculty of Computing and Information Technology", "Students are trained in both theoretical
knowledge and practical skills for software development, system design and related
mathematical techniques."));
128         pList.add(new Program("DIT", "Diploma", "Diploma in Information Technology",
"FOCS - Faculty of Computing and Information Technology", "This programme provides students
with a basic understanding of computing techniques and aims to develop the computing and
information technology-based knowledge and skills required in modern industrial, commercial
and service organisations."));
129         pList.add(new Program("DIS", "Diploma", "Diploma in Information System", "FOCS -
Faculty of Computing and Information Technology", "This programme majors in business
information systems. It aims to produce graduates with fundamental knowledge in information
technology and its business related applications."));
130         pList.add(new Program("RCB", "Bachelor Degree", "Bachelor of Communication
(Honours) in Broadcasting", "FCCI - Faculty of Communication & Creative Industries", "This
```

```

programme entails knowledge and skills for the technical understanding, writing, scripting,
presenting and organising of traditional and new broadcast media."));
131         pList.add(new Program("RFS", "Bachelor Degree", "Bachelor of Science (Hons) in
Food Science", "FOAS - Faculty of Applied Sciences", "This programme applies the pure
science subjects, such as chemistry, biochemistry, nutrition, biology and microbiology to
the study of the nature, properties and composition of foods."));
132         pList.add(new Program("RCA", "Bachelor Degree", "Bachelor of Corporate
Administration (Honours)", "FAFB - Faculty of Accountancy, Finance & Business", "This
programme provides students with broad base knowledge in management, law, accounting,
taxation, finance and corporate governance."));
133         pList.add(new Program("RSW", "Bachelor Degree", "Bachelor of Software
Engineering (Honours)", "FOCS - Faculty of Computing and Information Technology", "Graduates
of this programme will be able to develop, manage and maintain high-quality software in a
systematic, controlled and efficient manner. "));
134         return pList;
135     }
136
137     public static void inputProgram() {
138         Initializer p = new Initializer();
139         ListInterface<Program> programList = p.ProgramInitializer();
140
141         // Save the program data to the file using ProgramDAO
142         DAO DAO = new DAO();
143         DAO.saveToFile(programList, "program.dat");
144
145         System.out.println("\nProgram: \n" + programList);
146     }
147
148     // Method to return a collection of with hard-coded entity values
149     public ListInterface<TutorialProgram> TutorialPrInitializer() {
150         ListInterface<TutorialProgram> tpList = new ArrList<>();
151         tpList.add(new TutorialProgram("RSD", "Bachelor of Information Technology
(Honours) in Software Systems Development", "RSDG1", 24, "", 2023, 6));
152         tpList.add(new TutorialProgram("RBA", "Bachelor of Business Administration
(Honours)", "RBAG1", 25, "", 2023, 6));
153         tpList.add(new TutorialProgram("RAC", "Bachelor of Accounting (Honours)",
"RACG1", 23, " ", 2023, 1));
154         tpList.add(new TutorialProgram("RBF", "Bachelor of Banking and Finance
(Honours)", "RBFG1", 24, "Chou Tzuyu", 2023, 6));
155         tpList.add(new TutorialProgram("RME", "Bachelor of Mechanical Engineering with
Honours", "RMEG1", 23, "Zhao LuSi", 2023, 6));
156
157         return tpList;
158     }
159
160     public static void inputTutorialProgram() {

```

```
161     Initializer tp = new Initializer();
162     ListInterface<TutorialProgram> tpList = tp.TutorialPrInitializer();
163
164     // Save the program data to the file using ProgramDAO
165     DAO DAO = new DAO();
166     DAO.saveToFile(tpList, "tutorialProgram.dat");
167
168     System.out.println("\nProgram: \n" + tpList);
169 }
170
171 }
```