**UECS2344 Software Design**

**Practical 6 Answer**

Iteration 1

1.

(a) Package Diagram

courses.app

ConsoleUI

CoursesApp

courses.domain

Course

<<interface>>

IDataStore

Controller

DataLists

(b) Design Class Diagram

**«boundary»**

ConsoleUI

**«control»**

Controller

**«entity»**

Course

- controller: Controller

+ addCourse(title: String, code: String)

+ displayAllCourses()

- dataLists: IDataStore

+ addCourse(title: String, code: String)

+ getNumberOfCourses(): int

+ getAllCourses(): List<Course>

- title: String

- code: String

1

\*

+ <<cerate>> Course(title: String,

code: String)

+ getTitle(): String

+ getCode(): String

DataLists

- courses: List<Course>

1

+ addCourse(course: Course)

+ getNumberOfCourses(): int

+ getAllCourses(): List<Course>

**«interface»**

IDataStore

+ addCourse(course: Course)

+ getNumberOfCourses(): int

+ getAllCourses(): List<Course>

(c) Code

|  |
| --- |
| **package** courses.domain;  **public** **class** Course {  **private** String title;  **private** String code;  **public** Course(String title, String code) {  **this**.title = title;  **this**.code = code;  }  **public** String getTitle() {  **return** title;  }  **public** String getCode() {  **return** code;  }  } |
| **package** courses.domain;  **import** java.util.List;  **public** **interface** IDataStore {  **public** **void** addCourse(Course course);    **public** **int** getNumberOfCourses();  **public** List<Course> getAllCourses();  } |
| **package** courses.domain;  **import** java.util.List;  **import** java.util.ArrayList;  **public** **class** DataLists **implements** IDataStore {  **private** List<Course> courses;    **public** DataLists() {  courses = **new** ArrayList<Course>();  }    **public** **void** addCourse(Course aCourse) {  courses.add(aCourse);  }    **public** **int** getNumberOfCourses() {  **return** courses.size();  }  **public** List<Course> getAllCourses() {  **return** courses;  }  } |
| **package** courses.domain;  **import** java.util.List;  **public** **class** Controller {  **private** IDataStore dataLists;  **public** Controller(IDataStore dataLists) {  **this**.dataLists = dataLists;  }  **public** **void** addCourse(String title, String code) {  Course newCourse = **new** Course(title, code);  dataLists.addCourse(newCourse);  }  **public** **int** getNumberOfCourses() {  **return** dataLists.getNumberOfCourses();  }  **public** List<Course> getAllCourses() {  **return** dataLists.getAllCourses();  }  } |

Iteration 2

2.

(a) Modified Design Class Diagram

**«boundary»**

ConsoleUI

**«control»**

Controller

**«entity»**

Course

- controller: Controller

+ addCourse(title: String, code: String)

+ displayAllCourses()

+ selectCourse(code: String): String

+ checkStudentInCourse(name: String) : String

+ enrollStudent(name: String, id: Sting)

- dataLists: IDataStore

+ addCourse(title: String, code: String)

+ getNumberOfCourses(): int

+ getAllCourses(): List<Course>

+ selectCourse(code: String): Course

+ checkStudentInCourse(name: String) : Boolean

+ enrollStudent(selectedCourse: Course,

name: String, id: int)

- title: String

- code: String

- studentsEnrolled: List<Student>

1

\*

+ <<cerate>> Course(title: String,

code: String)

+ getTitle(): String

+ getCode(): String

+ isStudentEnrolled(name: String) :

boolean

+ enrollStudent(student: Student)

DataLists

- courses: List<Course>

1

+ addCourse(course: Course)

+ getNumberOfCourses(): int

+ getAllCourses(): List<Course>

+ searchCourse(code: String): Course

**«interface»**

IDataStore

**«entity»**

Student

- name: String

- id: int

+ <<cerate>> Student(name: String, id: int)

+ getName(): String

+ getId(): int

+ addCourse(course: Course)

+ getNumberOfCourses(): int

+ getAllCourses(): List<Course>

+ searchCourse(code: String): Course

\*

(b) Modified Code

|  |
| --- |
| **package** courses.domain;  **import** java.util.List;  **import** java.util.ArrayList;  **public** **class** Course {  **private** String title;  **private** String code;  **private** List<Student> studentsEnrolled;  **public** Course(String title, String code) {  **this**.title = title;  **this**.code = code;  studentsEnrolled = **new** ArrayList<Student>();  }  **public** String getTitle() {  **return** title;  }  **public** String getCode() {  **return** code;  }  **public** **boolean** isStudentEnrolled(String name) {  **int** size = studentsEnrolled.size();  **boolean** isEnrolled = **false**;  **int** i = 0;  Student aStudent = **null**;  **while** (i<size && !isEnrolled) {  aStudent = studentsEnrolled.get(i);  **if** (aStudent.getName().equals(name))  isEnrolled = **true**;  **else**  i++;  }  **return** isEnrolled;  }  **public** **void** enroll(Student aStudent) {  studentsEnrolled.add(aStudent);  }  } |
| **package** courses.domain;  **public** **class** Student {  **private** String name;    **private** **int** id;  **public** Student(String name, **int** id) {  **this**.name = name;  **this**.id = id;  }  **public** String getName() {  **return** name;  }    **public** **int** getId() {  **return** id;  }  } |
| **package** courses.domain;  **import** java.util.List;  **public** **interface** IDataStore {  **public** **void** addCourse(Course course);    **public** **int** getNumberOfCourses();  **public** List<Course> getAllCourses();  **public** Course searchCourse(String code);  } |
| **package** courses.domain;  **import** java.util.List;  **import** java.util.ArrayList;  **public** **class** DataLists **implements** IDataStore {  **private** List<Course> courses;    **public** DataLists() {  courses = **new** ArrayList<Course>();  }    **public** **void** addCourse(Course aCourse) {  courses.add(aCourse);  }    **public** **int** getNumberOfCourses() {  **return** courses.size();  }  **public** List<Course> getAllCourses() {  **return** courses;  }  **public** Course searchCourse(String code) {  Course theCourse = **null**;  **boolean** found = **false**;  **int** i = 0;  **while** (i<courses.size() && !found) {  theCourse = courses.get(i);  **if** (theCourse.getCode().equals(code))  found = **true**;  **else**  i++;  }  **if** (!found)  theCourse = **null**;  **return** theCourse;  }  } |
| **package** courses.domain;  **import** java.util.List;  **public** **class** Controller {  **private** IDataStore dataLists;  **public** Controller(IDataStore dataLists) {  **this**.dataLists = dataLists;  }  **public** **void** addCourse(String title, String code) {  Course course = **new** Course(title, code);  dataLists.addCourse(course);  }  **public** **int** getNumberOfCourses() {  **return** dataLists.getNumberOfCourses();  }  **public** List<Course> getAllCourses() {  **return** dataLists.getAllCourses();  }  **public** Course selectCourse(String code) {  **return** dataLists.searchCourse(code);  }    **public** **boolean** checkStudentInCourse(Course selectedCourse,  String name) {  **return** selectedCourse.isStudentEnrolled(name);  }  **public** **void** enrollStudent(Course selectedCourse, String name,  **int** id) {  Student aStudent = **new** Student(name, id);  selectedCourse.enroll(aStudent);  }  } |
| **package** courses.app;  **import** java.util.List;  **import** java.util.Scanner;  **import** courses.domain.\*;  **public** **class** ConsoleUI {  **private** Scanner scanner;  **private** Controller controller;  **public** ConsoleUI(Controller controller) {  scanner = **new** Scanner(System.***in***);  **this**.controller = controller;  }  **public** **void** start() {  **int** choice;  **do** {  System.***out***.println("Do you want to:");  System.***out***.println("1. Display all courses");  System.***out***.println("2. Add new course");  System.***out***.println("3. Check if student enrolled in course");  System.***out***.println("4. Enroll student in course");  System.***out***.println("5. Exit");  System.***out***.print("Enter your choice (1-5): ");  choice = scanner.nextInt();  // Clear ENTER key after integer input  String skip = scanner.nextLine();  **while** (choice < 1 || choice > 5) {  System.***out***.println("Invalid choice.");  System.***out***.print("Enter your choice (1-6): ");  choice = scanner.nextInt();  // Clear ENTER key after integer input  skip = scanner.nextLine();  }  **switch**(choice) {  **case** 1: displayAllCourses(); **break**;  **case** 2: addCourse(); **break**;  **case** 3: checkStudentInCourse(); **break**;  **case** 4: enrollStudent(); **break**;  **case** 5: **break**;  }  System.***out***.println();  } **while** (choice != 5);  }  **public** **void** displayAllCourses() {  **int** count = controller.getNumberOfCourses();  **if** (count == 0)  System.***out***.println("No courses to display");  **else** {  List<Course> theCourses = controller.getAllCourses();  Course aCourse;  **for** (**int** i=0; i<count; i++) {  aCourse = theCourses.get(i);  System.***out***.println("Code: " + aCourse.getCode()  + "\tTitle: " + aCourse.getTitle());  }  }  }  **public** **void** addCourse() {  System.***out***.print("Enter course title: ");  String theTitle = scanner.nextLine();  System.***out***.print("Enter course code: ");  String theCode = scanner.nextLine();  controller.addCourse(theTitle, theCode);  System.***out***.println("New course created");  System.***out***.println();  }  **private** Course selectCourse() {  System.***out***.print("Enter course code: ");  String theCode = scanner.nextLine();  Course selectedCourse = controller.searchCourse(theCode);  **if** (selectedCourse == **null**)  System.***out***.println("No course with code " + theCode  + " found");  **else**  System.***out***.println("Title: "  + selectedCourse.getTitle());  **return** selectedCourse;  }  **public** **void** checkStudentInCourse() {  Course selectedCourse = selectCourse();  **if** (selectedCourse != **null**) {  System.***out***.print("Enter stduent name: ");  String name = scanner.nextLine();  **if** (controller.checkStudentInCourse(selectedCourse,  name))  System.***out***.println(name + " is enrolled in course");  **else**  System.***out***.println(name + " is not enrolled in course");  }  }  **public** **void** enrollStudent() {  Course selectedCourse = selectCourse();  **if** (selectedCourse != **null**) {  System.***out***.print("Enter stduent name: ");  String name = scanner.nextLine();  System.***out***.print("Enter stduent id: ");  **int** id = scanner.nextInt();  controller.enrollStudent(selectedCourse, name, id);  System.***out***.println("Student enrolled in course");  }  System.***out***.println();  }  } |
| **package** courses.app;  **import** courses.domain.\*;  **public** **class** CoursesApp {    **public** **static** **void** main(String[] args) {  IDataStore dataLists = **new** DataLists();    Controller controller = **new** Controller();  controller.setCourseList(dataLists);    ConsoleUI userInterface = **new** ConsoleUI();  userInterface.setController(controller);  userInterface.start();    }  } |