Name:Ma Phyo Zin Mar

***Course Registration Online System***

**Entity**

1. Student.java

Declare studentId, name,email,password and mobile ,generate getter and setter for this. Print “Student Information” in the toString ( ).

1. Registration.java

Declare id,course,student,semester,year and status.And then generate getter and setter methods for this .

1. Course.java

Declare id,name and department.And then generate getter and setter methods for this.Print “Course Information” in the toString().

1. Department.java

Declare id,name and phone.And then generate getter and setter methods for this.

**Repository**

1.StudentRepository.java

Create LinkList to store Student informations.Generate getter and setter for these LinkList Student.Construct student object naming defaultuser and jaya.Insert email,mobile,name and password into defaultuser and jaya by using “Set” method.And then add to the linklist student.Then write two methods ‘getUserById ‘by checking ‘id’ and ‘getUserByEmail’by checking‘email’.

2.RegistrationRepository.java

Create LinkList to store Registration informations.Generate getter and setter for these LinkList Registration.Create getRegistrationByStudentId method by checking ‘id’ and return type is “List<registration>.Create getById method by checking ‘id’ and return type is Registration.

3.CouseRepository.java

Create Array list naming course.Construct DepartmentRepository object and adding id 1 assign to sanskirtDepartment and id 2 assign to historyDepartment.Then department,id and name are added into course.Generate getter and setter method for course .Write getById method argument is id, return type is course and findByDepartment argument is department, return type is List<course>.

4.DepartmentRepositoty.java

Create Linklist naming department. Construct two objects naming sanskirtDepartment and historyDepartment.Id,name and phone are added into these objects by using setMethod and then finally added into linklist department.Generate getter and setter methods for department.Also create findbyId method arguments is id and return type is department.

**Service**

1.StudentService.java

Call studentRepository and build default constructor for studentService.Write getUser () and return type is student list.Then write setUser() and argument is student.Write register () and inserts five parameters.Construct student object , add value to object ,add object to repository.setuser and return type is object.And then write login() with two paramenters and add existing email in repository to student object .Check with if condition and existing password and login password are equal or not equal.If not equal return incorrectpasswordexception.

2.RegistrationService.java

Call repositories except department repository, construct default constructor and built all objects in default constructor.Write getRegistrationByStudentId with paramenter is id and return type is registration list. Write getById () with parameter is id and return type is registration. Write save() with 4 parameters. Insert studentId to student object and insert courseId to course object. And construct registration object and set data into object .Then added to repository and return type is id.

3.CourseService.java

Call couresRepository and departmentRepository.Construct objects into default constructor .Create getCourseByDepartment with paramenter is id and return type is course list.Id from departmentRepository add to deparment object.Then department object into courseRepository.Create getCourseByDepartment() with parameter is sid and return type is course list.Convert string sid into integer and return getCourseByDepartment() .Use try-catch exception and condition is false return is new arraylist.Write getcourse() with return type is course list and write setcourse.

4.DepartmentService.java

Call department Repository and construct defalult constructor.Create getDepartment with return type is departmentlist.

**Exception**

*IncorrectPasswordException/departmentnotfound/invalidemailaddress/usernotfoundexception*

Exception entends RuntimeException and create one argument constructor.

**Filter**

studentfilter

Create studentfilter implement filter and write dofilter parameter with request, response and chain. Use if condition for email and password exist or not exist. If not exist ,nosuchElementException occur. Email and password are convert into string. Use if condition to check email and password are empty then condition is true ,illegalgument exception occur and email are not match ,invalidemailaddress exception occur.

Registerfilter

Create registerfilter implement filter and wirte dofilter parameter with request,response and chain.Use if condition for name,email,mobile,studentId ,password and confirmpassword exist or not exist.If not exist ,nosuchElementException occur.Name,email,mobile,studentId,password and confirmpassword are convert into string.Use if condition to check

,name,email,mobile,studentId,password and confirmpassword are empty then condition is true ,illegalargument exception occour and email are no matched,invalidemailaddress exception occur.Password are not matched ,incorrectpasswordexception occur.

CourseRegisterfilter

Create courseregisterfilter implement filter and write dofilter parameter with request, response and chain. Use if condition for courseId,semester and year exist or not exist. If not exist ,nosuchElementException occur. CourseId,semester and year are convert into string. Use if condition to check courseId,semester and year are empty then condition is true ,illegalargument exception occur .

**Controller**

Userregisterservlet

userRegisterServlet extends HttpServlet and call studentService.Construct studentService object in init().Write dopost() method and email,mobile,name,studentId and password form textbox convert into string.Then register into stuentService.Response to request contexttpath.

UserLoginservlet

userRegisterServlet extends HttpServlet and call studentService.Construct studentService object in init().Write doGet() method and email,password and student form textbox convert into string.Use session and add to student.Response to request contexttpath dashboard.jsp.

Userlogoutservlet

userlogoutServlet extends HttpServlet.Write doGet() method and use session.invaliddate. .Response to request contexttpath.

Userservlet

userServlet extends HttpServlet and call studentService.Construct studentService object in init().Write doGet() method and insert student service’s data into iterator and print.

CourseRegisterServlet

courseRegisterServlet extends HttpServlet and call RegistrationService.Construct registrationService object in init().Write dopost() method and if user is null ,go to index.jsp.StudentId from user convert into long.CourseId,semester and year from textbox into string .Then data are saved into service .Response to confirm.jsp.