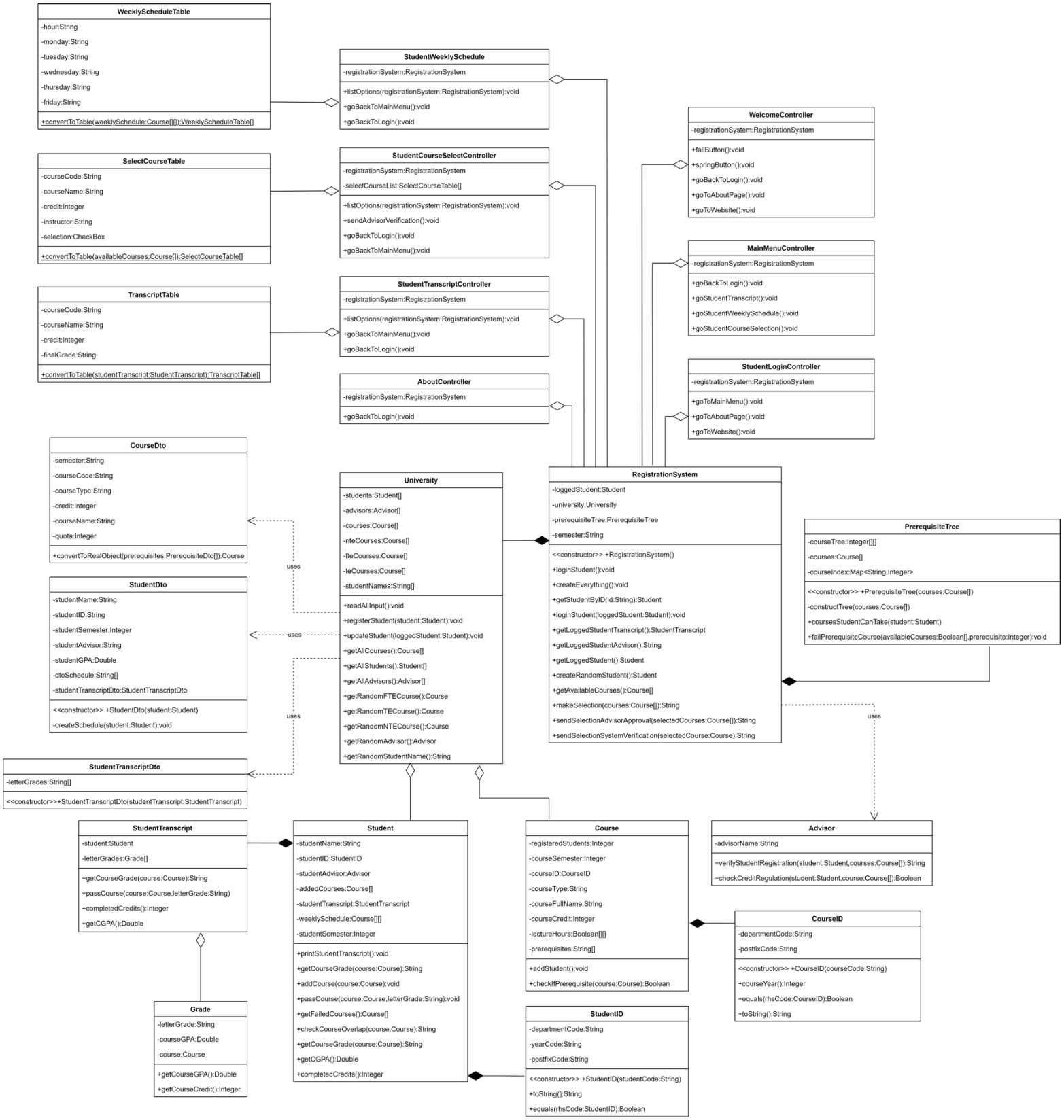
WeeklyScheduleTable
-hour:String
-monday:String
-tuesday:String
-wednesday:String
-thursday:String
-friday:String
+convertToTable(weeklySchedule:Course[][]):WeeklyScheduleTable[]
StudentWeeklySchedule
-registrationSystem:RegistrationSystem
+listOptions(registrationSystem:RegistrationSystem):void
+goBackToMainMenu():void
+goBackToLogin():void
WelcomeController
-registrationSystem:RegistrationSystem
+fallButton():void
+springButton():void
+goBackToLogin():void
+goToAboutPage():void
+goToWebsite():void
SelectCourseTable
-courseCode:String
-courseName:String
-credit:Integer
-instructor:String
-selection:CheckBox
+convertToTable(availableCourses:Course[]):SelectCourseTable[]
StudentCourseSelectController
-registrationSystem:RegistrationSystem
-selectCourseList:SelectCourseTable[]
+listOptions(registrationSystem:RegistrationSystem):void
+sendAdvisorVerification():void
+goBackToLogin():void
+goBackToMainMenu():void
MainMenuController
-registrationSystem:RegistrationSystem
+goBackToLogin():void
+goStudentTranscript():void
+goStudentWeeklySchedule():void
+goStudentCourseSelection():void
TranscriptTable
-courseCode:String
-courseName:String
-credit:Integer
-finalGrade:String
+convertToTable(studentTranscript:StudentTranscript):TranscriptTable[]
StudentTranscriptController
-registrationSystem:RegistrationSystem
+listOptions(registrationSystem:RegistrationSystem):void
+goBackToMainMenu():void
+goBackToLogin():void
StudentLoginController
-registrationSystem:RegistrationSystem
+goToMainMenu():void
+goToAboutPage():void
+goToWebsite():void
AboutController
-registrationSystem:RegistrationSystem
+goBackToLogin():void
CourseDto
-semester:String
-courseCode:String
-courseType:String
-credit:Integer
-courseName:String
-quota:Integer
+convertToRealObject(prerequisites:PrerequisiteDto[]):Course
uses
University
-students:Student[]
-advisors:Advisor[]
-courses:Course[]
-nteCourses:Course[]
-fteCourses:Course[]
-teCourses:Course[]
-studentNames:String[]
+readAllInput():void
+registerStudent(student:Student):void
+updateStudent(loggedStudent:Student):void
+getAllCourses():Course[]
+getAllStudents():Student[]
+getAllAdvisors():Advisor[]
+getRandomFTECourse():Course
+getRandomTECourse():Course
+getRandomNTECourse():Course
+getRandomAdvisor():Advisor
+getRandomStudentName():String
RegistrationSystem
-loggedStudent:Student
-university:University
-prerequisiteTree:PrerequisiteTree
-semester:String
<<constructor>> +RegistrationSystem()
+loginStudent():void
+createEverything():void
+getStudentByID(id:String):Student
+loginStudent(loggedStudent:Student):void
+getLoggedStudentTranscript():StudentTranscript
+getLoggedStudentAdvisor():String
+getLoggedStudent():Student
+createRandomStudent():Student
+getAvailableCourses():Course[]
+makeSelection(courses:Course[]):String
+sendSelectionAdvisorApproval(selectedCourses:Course[]):String
+sendSelectionSystemVerification(selectedCourse:Course):String
PrerequisiteTree
-courseTree:Integer[][]
-courses:Course[]
-courseIndex:Map<String,Integer>
<<constructor>> +PrerequisiteTree(courses:Course[])
-constructTree(courses:Course[])
+coursesStudentCanTake(student:Student)
+failPrerequisiteCourse(availableCourses:Boolean[],prerequisite:Integer):void
StudentDto
-studentName:String
-studentID:String
-studentSemester:Integer
-studentAdvisor:String
-studentGPA:Double
-dtoSchedule:String[]
-studentTranscriptDto:StudentTranscriptDto
<<constructor>> +StudentDto(student:Student)
-createSchedule(student:Student):void
uses
StudentTranscriptDto
-letterGrades:String[]
<<constructor>>+StudentTranscriptDto(studentTranscript:StudentTranscript)
uses
StudentTranscript
-student:Student
-letterGrades:Grade[]
+getCourseGrade(course:Course):String
+passCourse(course:Course,letterGrade:String)
+completedCredits():Integer
+getCGPA():Double
Grade
-letterGrade:String
-courseGPA:Double
-course:Course
+getCourseGPA():Double
+getCourseCredit():Integer
Student
-studentName:String
-studentID:StudentID
-studentAdvisor:Advisor
-addedCourses:Course[]
-studentTranscript:StudentTranscript
-weeklySchedule:Course[][]
-studentSemester:Integer
+printStudentTranscript():void
+getCourseGrade(course:Course):String
+addCourse(course:Course):void
+passCourse(course:Course,letterGrade:String):void
+getFailedCourses():Course[]
+checkCourseOverlap(course:Course):String
+getCourseGrade(course:Course):String
+getCGPA():Double
+completedCredits():Integer
Course
-registeredStudents:Integer
-courseSemester:Integer
-courseID:CourseID
-courseType:String
-courseFullName:String
-courseCredit:Integer
-lectureHours:Boolean[][]
-prerequisites:String[]
+addStudent():void
+checkIfPrerequisite(course:Course):Boolean
Advisor
-advisorName:String
+verifyStudentRegistration(student:Student,courses:Course[]):String
+checkCreditRegulation(student:Student,course:Course[]):Boolean
CourseID
-departmentCode:String
-postfixCode:String
<<constructor>> +CourseID(courseCode:String)
+courseYear():Integer
+equals(rhsCode:CourseID):Boolean
+toString():String
StudentID
-departmentCode:String
-yearCode:String
-postfixCode:String
<<constructor>> +StudentID(studentCode:String)
+toString():String
+equals(rhsCode:StudentID):Boolean