Das\_teamCS374

Shawn, Steven, Ivan

Software Engineering

Dr. Reeves

09/27/2016

**Glossary**

* File CourseTest.java:
  + CourseTest: the main class that contain all the steps.
  + setup(): initialize the new entity
  + thatStudentWantsToTakeTheCourse(): calling function getApproval() to insert data from
  + itIsResultThatStudentStudentIsAbleToTakeTheCourse(): a method to check if the Student is put in the class correctly.
  + aProfessorWantsToFindAStudentAndTheTotalNumberOfClassesHeHasTaken(String arg1, String arg2)—This function will how many class a student has taken.
  + theNumberOfClasses—For checking
  + aProfessorWantsToFindHowManyStudentsTookClassSectionClassName(int arg1, String arg2, String arg3)—A professor wants to check how many students has taken a clas.
  + theNumberOfStudents(int arg1)—For checking.
  + aProfessorWantsToKnowIfAListOfStudentsAreCapableToTakeAClass(String arg1, String arg2, int arg3, String arg4, int arg5)—check who is not capable to take a class from a list of students.
  + theClassHasAPreq\_list(String arg1)—Get the course preQ list.
  + aStudentHasTakenTheseListOfClasses(String arg1, String arg2)—Get the students class records.
  + theShouldBeInThisClass(String arg1)—Check Results.
* File CSVParse.java
  + CVSParse – A class that contains method that parses data from the CSV file
  + getValuesFromCSV() – A method gets values from a specified CSV file
  + notInClass()- A method to check if the student has taken the course or not.
* File GetStudents.java
  + GetStudents – A class with methods that handles different student cases
  + checkCourse(): a method to split input from CSV, add data to the correct vector.
  + getApproval(): a method to check if the student meets the course requirement.
  + getHours() – A method that returns the total hours
  + getNumberOfCourses () – A method that returns the total number of classes a student is taking
  + compare() – A method that checks if a student has met the prerequisites for a specific class
  + getName() – A method that returns the name of a student
  + getResult() -- A method that returns the boolean string variable “iAMok” if the results are positive
  + addTakenCourses() -- A method that saves the parameter value into a vector
  + currentClass() – Set input as the class the student is taking
  + takeStudent(String first, String last) – get access for MySQLAccess. Return the number of classes this student is taking.
  + getCount() – this convert the String output from takeStudent() into a integer.
  + listOfStudents – This is a 2D array of student names.
  + listOfPreReq—This is a 2D array of pre-requisite of courses.
  + listOfStudentInfo—This is a Dictionary of Student names and class information
  + takeListOfStudents(List<List<String>> students)—This function will insert data into listOfStudentInfo.
  + getPreReqData()—This function will get preQ for each class.
* MySQLAccess.java
  + getStudentCount(String, String) – Set up the connection with database on local device. Parse sql query and get count for how many classes a student has taken. Store count into a local String.
  + resultSet – execute SQL query
  + internal = new ArrayList<String>
  + while (resultSet.next()) – while resultSet is not null, store output from SQL into internal.
  + listOfLists = new ArrayList<List<String>>() : this is a table of lists holds all the data from internal.
  + writeResultSet(ResultSet resultSet) – while resultSet is not null, print all the data it received onto the screen.
  + getPreReqData()—Get preQ from MySQL.
  + getStudentInfo(String firstName, String lastName)—Get student information list from MySQL.
  + close() – close Resultset, connection, statement, from java package, and disconnect database.