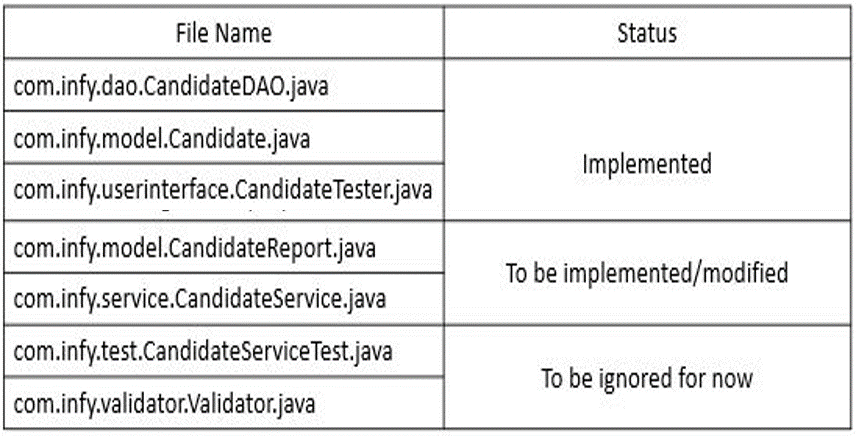
​​​​​​​**ABCAcademy**is an online academy which invites candidates to participate in various events, and also provides grades for each candidate based on his/her performance. They need your help in two modules,

* Adding candidates
* Getting candidate reports

**Artifacts:**



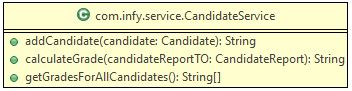
**CandidateReport:**

Please implement the class based on the class diagram given below. Please follow the class diagram strictly.



Make sure that you keep the variable names as it is shown in the class diagram.

**CandidateService:**



This class has three methods which you have to implement based on the description given below,

**addCandidate(Candidate candidate):**

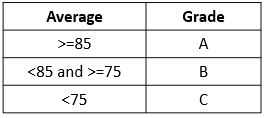
This method accepts a Candidate object, validates the marks and result of the candidate and sends it to the CandidateDAO class to be inserted to the database table.

* If any one of the marks attribute of the Candidate object is below 50, and the result attribute of the Candidate object is '**P'**, return the String message- "**Result should be 'F' (Fail) if student scores less than 50 in any one subject".**
* Else, invoke the addCandidate() from CandidateDAO class by passing the Candidate object as the parameter, which will return a String value.
* Return the String value obtained in the previous step.

**calculateGrade(CandidateReport candidateReportTO):**

This method calculates the average of the marks in the CandidateReport object, and return a grade relevant to the average.

* If the result of the CandidateReport object is **F**, return **NA**
* Else, calculate the average of the marks attribute of the CandidateReport object.
* Return the grade based on the average obtained in the previous according to the table given below,



**getGradesForAllCandidates():**

This method gets all the CandidateReport details from the CandidateDAO, creates a String value based on the CandidateReport object's candidateID and the grade obtained by invoking the calculateGrade() method of the CandidateService class.

* Invoke the getAllCandidates() method from the CandidateDAO class which will return a CandidateReport array.
* Iterate over the array obtained in the previous step and create a String value for each of the object in the below format,

**candidateId:grade**

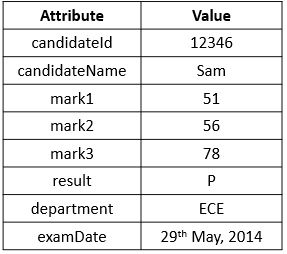
where, **grade** is obtained by invoking the calculateGrade() method of the CandidateService class by passing the CandidateReport object as the parameter.

* Add all the created String values in a new String array.
* Return array created in the previous step.

**Sample Input and Output:**

**Add Candidate**

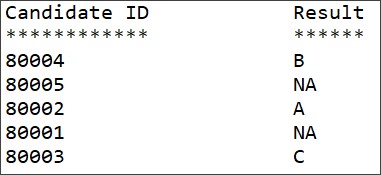
* **Input:**



* **Output:** Candidate details are successfully added.

**Get Candidates Results:**

* **Output:**



**Note**: Check the project using SonarLint to maintain the coding standards. Ignore the violations which occur due to "System.out" statements.