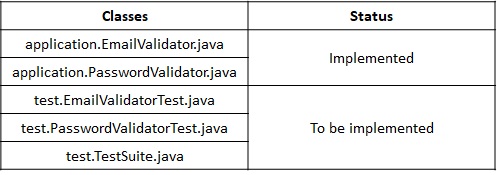
**Problem Statement**

The Golden Trio Ltd wants to hire new and potential individuals to their company. For that, they have created an application which will accept the details of the individual and check if the details are correct. If they are, then the individuals will be called upon for an interview. But they are facing issues with a part of their code. Hence they have asked you to test a few snippets of their project.

**Artifacts:**



**EmailValidator class:**

https://academy.onwingspan.com/common-content-store/Shared/Shared/Public/lex_auth_0130080565919744001159_shared/web-hosted/assets/JUnitExercise1.JPG

This class is already implemented and has one method,

**validateEmailId(String emailId):**

* This method validates the received *emailId*.
* If the emailId is null or blank, an exception will be thrown with the message, *Invalid Email ID.*
* Else, it checks if the emailId follows the following format,
  + **<<Part1>>@<<Part2>>.<<Part3>>**, where
  + Part 1 & Part 2 - Any word character (alphanumeric with an underscore)
  + Part 3 - Either *com* or *in*
* If the above condition fails, then it returns false, else it returns true.

**PasswordValidator:**

https://academy.onwingspan.com/common-content-store/Shared/Shared/Public/lex_auth_0130080565919744001159_shared/web-hosted/assets/JUnitExercise2.JPG

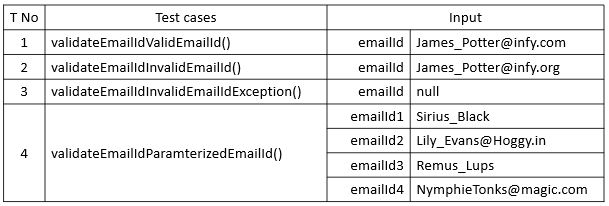
This class is already implemented and has one method,

**validatePassword(String password):**

* This method validates the received *password*.
* If the *password*is null or blank, an exception will be thrown with the message, *Invalid Password.*
* The password should contain only alphanumeric values with a length of minimum 8 and maximum 20.
* If the above condition fails, then it returns false, else it returns true.

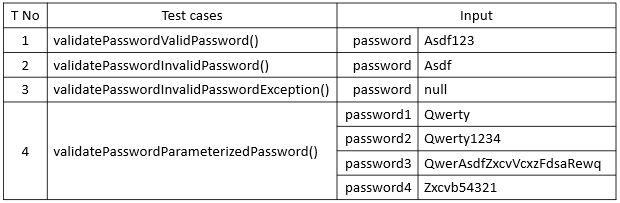
**EmailValidatorTest class:**

This is the test class that you will have to implement. This class test the method of the EmailValidator class. The class and the methods have been created for you. You have to annotate each method properly and proceed with creating the test cases. The inputs for each method have been given to you in the table below. Please observe the input provided, and devise the expected output based on the description given above.



**PasswordValidatorTest class:**

This is the test class that you will have to implement. This class tests the methods of the PasswordValidator class. The class and the methods have been created for you. You have to annotate each method properly and proceed with creating the test cases. The inputs for each method have been given to you in the table below. Please observe the input provided, and devise the expected output based on the description given above.



**TestSuite class:**

Annotate this class with appropriate annotations so that the execution of this class will lead to the execution of the EmailValidatorTest class and the PasswordValidator test class automatically.

**Note**: Check the project using SonarLint to maintain the coding standards.