**Name**: Emma Mason

**Date:** 27/11/20

**Class Name and Description**:

Read class. A class to retrieve user input with exception handling. Should the user enter a null value or an empty string, they will be asked to renter this information again to ensure input.

|  |  |
| --- | --- |
| **Program Design** | **Data Requirements** |
| * A String method to read in a value from the user. Can be changed to an int with ParseInt method. * A print statement with a general prompt will be used. * Try and catch method within a while loop to ensure user input (exception handling) * String method will return the value user has entered. | * String value for user input. |

**Name**: Emma Mason

**Date:** 27/11/20

**Class Name and Description**:

RegistrationTransaction class. A class to register a new user to the application and store their password and username. Will distinguish between full time and casual user during registration.

|  |  |
| --- | --- |
| **Program Design** | **Data Requirements** |
| * Prompt user for information included in the User class – username, password, Instagram handle and their total followers. * Will then ask user for their type – full time or casual. * If statement to filter the users to their appropriate objects – full time or user. * User will be returned. | * User object for default. * String username, password, userAt. * Int totalFollowers. * String userType. |

**Name**: Emma Mason

**Date:** 27/11/20

**Class Name and Description**:

UserLoginTransaction class. A class to allow for login of hard coded users and newly registered users.

|  |  |
| --- | --- |
| **Program Design** | **Data Requirements** |
| * Read in username and password using read method. * Use a for loop o count through the entry. * Return user not found if it does not match. * Return true if user is found. | * Boolean successfulLogin. * Int index. * An array list – List<User> users. * String username, password. * Int pos. |

**Name**: Emma Mason

**Date:** 18/11/20

**Class Name and Description**:

Casual User, a sub class of class user. A class to store the variables and methods specific to a casual Instagram user. These methods will include constructors, getters and setters for all the instance variables, a calculation method for user interaction, a method to calculate their reach grade, a method to return tips and print and update methods for certain variables.

|  |  |
| --- | --- |
| **Program Design** | **Data Requirements** |
| * Declare instance variables specific for a casual user. * Constructors for the casual user objects * Getters and setters for the instance variable * Method to calculate follower’s interaction with the user * Method to calculate the user’s reach * Method to check a user’s reach score. If low, give the user improvement tips.. * Print methods. * To string method for all user information | * Super from user class of username, password, Instagram handle and their total followers as well as int variables of the user’s latest three likes – int likes1, likes2, likes3. * Users weekly Instagram posts – int postsPerWeek. * Users daily Instagram stories – int storiesPerDay. * Users interaction percentage – double interactPercentage. * A grade calculated based on interaction – char grade. * A String array to store various tips on when to post on Instagram – String [] timeimprovement. |