Homework 02
Rock Paper Scissors!

Objective:

Write a program that simulates a game of rock, paper, scissors between a human and a computer opponent in best 2 out of 3 rounds.

Requirements:
- Functionality. (80pts)
    - No Syntax, Major Run-Time, or Major Logic Errors. (80pts*)
        - *Code that cannot be compiled due to syntax errors is nonfunctional code and will receive no points for this entire section.
        - *Code that cannot be executed or tested due to major run-time or logic errors is nonfunctional code and will receive no points for this entire section.
    - Clear and Easy-To-Use Interface. (10pts)
        - Users should easily understand what the program does and how to use it.
        - Users should be prompted for input and should be able to enter data easily.
        - Users should be presented with output after major functions, operations, or calculations.
        - All the above must apply for full credit.
    - Determine the Winner of a Round. (20pts)
        - The user must select either "Rock", "Paper", or "Scissors".
        - The computer opponent must randomly select either "Rock", "Paper", or "Scissors" each round.
        - A winner must be determined by examining the user's input and the computer's selection.
        - Rock vs Paper = Paper Wins.
        - Paper vs Scissors = Scissors Wins.
        - Scissors vs Rock = Rock Wins.
        - Each winner receives 1 point.
        - In the event of a tie there is no winner, and no points are awarded.
        - If the user enters an incorrect value, then the computer automatically wins that round and receives a point.
        - The results of each round must be displayed – either a winner or a tie.
        - All must apply for full credit.
    - Declare an overall Winner or a Tie. (20pts)
        - The program must determine and display an overall winner after exactly 3 rounds via their points.
        - In the event of a tie, then the program must display that the user and the computer tied.

- All must apply for full credit.
  - Replay the Game. (20pts)
    - Once the 3 rounds have concluded, the program must ask the user if they would like to play again.
    - If the user answers yes, then the scores must be reset and the game restarts.
    - Otherwise, the program must terminate.
- Coding Style. (10pts)
  - Readable Code
    - Meaningful identifiers for data and methods.
    - Proper indentation that clearly identifies statements within the body of a class, a method, a branching statement, a loop statement, etc.
    - All the above must apply for full credit.
- Comments. (10pts)
  - Your name in the file. (5pts)
  - At least 5 meaningful comments in addition to your name. These must describe the function of the code it is near. (5pts)
- **Extra credit (10 pts)**
  - Create a program that <u>remembers</u> if a user always puts one of the three possibilities and exploits this during the next round.
  - For example, for the first round, the computer selects "Rock", "Paper", or "Scissors" randomly. However, if the user, for example, always selected "Rock" in the first round, then the computer will always select paper in the second round.
  - Do this <u>only</u> if the user always selects one choice.

Finally:
Upload the .java file to the CSCE Dropbox