Conditionals Challenge 20: Rock, Paper, Scissors App

Description:
You are responsible for writing a program that will simulate playing the classic game Rock, Paper, Scissors against computer AI. Your program will ask the user how many rounds of the game they would like to play, simulate each round, keep score, and determine an overall winner. Your program will also print the classic phrases such as "Paper covers rock" or "Scissors cut paper".

Step By Step Guide:
- Print a welcome message.
- Get user input for how many rounds they would like to play.

- Create a list called moves that holds three strings: 'rock', 'paper', and 'scissors'.
- Create a variable to hold the players score and set it to zero.
- Create a variable to hold the computer score and set it to zero.

- Play the game for the correct number of rounds using a for loop.
- Each round should do the following:
  - Print information for the current round.
    - This includes round number, player score, and computer score.
  - Have the computer randomly pick an element from the list moves.
    - Type import random as the first line of code in your program.
    - Randomly generate an integer from 0 to 2 which corresponds to an index in the list moves (0 = rock, 1 = paper, 2 = scissors).
    - Assign the computer a move based off this random index.
- Get the players choice of rock, paper, or scissors.
  - Make sure to make it case insensitive.

- If the player has made a valid move:
  - Print both moves.
  - Program the "logic" behind a game of rock, paper, scissors and play the round.
    - Rock beats scissors, scissors beats paper, paper beats rock.
    - For each outcome make sure you:
      - Set a round winner (player or computer).
      - Adjust the score.
      - Set a phrase (rock smashes scissors!) to be displayed.

- Else the player has not made a valid move:
  - Let the computer win the round.

- After all rounds have been played, print out the final game statistics and print a statement indicating who won the game.

- Use at least 2 comments to describe sections of your code.
- "Chunk" your code so that is readable.
- Use appropriate and informative variable names.
- Format your output as below.

Example Output 1:
Welcome to a game of Rock, Paper, Scissors

How many rounds would you like to play: 3

Round 1
Player: 0   Computer: 0
Time to pick...rock, paper, scissors: scissors
        Computer: paper
        Player: scissors
        Scissors cut paper!
        You win round 1.

Round 2
Player: 1   Computer: 0
Time to pick...rock, paper, scissors: PAPER
        Computer: rock
        Player: paper
        Paper covers rock!
        You win round 2.

Round 3
Player: 2   Computer: 0
Time to pick...rock, paper, scissors: Rock
        Computer: rock
        Player: rock
        It is a tie, how boring!
        This round was a tie.

Final Game Results
        Rounds Played: 3
        Player Score: 2
        Computer Score: 0
        Winner: PLAYER!!!

Example Output 2:
Welcome to a game of Rock, Paper, Scissors

How many rounds would you like to play: 3

Round 1

Player: 0   Computer: 0

Time to pick...rock, paper, scissors: pizza

That is not a valid game option!

Computer gets the point!

Round 2

Player: 0   Computer: 1

Time to pick...rock, paper, scissors: paper

        Computer: paper

        Player: paper

        It is a tie, how boring!

        This round was a tie.

Round 3

Player: 0   Computer: 1

Time to pick...rock, paper, scissors: Rock

        Computer: rock

        Player: rock

        It is a tie, how boring!

        This round was a tie.

Final Game Results

        Rounds Played: 3

        Player Score: 0

        Computer Score: 1

        Winner: Computer :-(