

Conditionals Challenge 16: Shipping Accounts Program

Description:

You are responsible for writing a program that will simulate logging into a business's shipping accounts software. Once logged in your program will display the current costs of shipping x amount of items. Based on the number of items shipped, the cost to ship each item will vary. Once the cost to ship an item is set, your program will calculate the cost of shipping the entire order. Upon confirmation of the order, your program will place the order and prepare the shipment.

Step By Step Guide:

- Begin by creating a list that will hold 5 user names of your choice.
 - For example: users = ['eramom', 'footea', 'davisv', 'papinukt', 'allenj']
- Print a welcome message to the user.
- Get user input for their specific username.
 - Usernames should not be case sensitive. For example, eramom == Eramom.
- If the username is in the list of usernames:
 - Welcome the user.
 - Print the shipping price summary as formatted below.
 - Get user input for how many items they would like to ship.
 - Use an if, elif, else chain to determine the cost of shipping each item.
 - Calculate the users bill based on the number of items to ship and the cost of shipping each item.
 - Round this bill to two decimal places.
 - Print the results as formatted below.
 - Ask the user if they would like to place their order.
 - If the users answer starts with y:
 - Print a message confirming that their order has been placed.
 - Else:
 - Inform the user that no order has been placed.
- Else, the user does not have an account:
 - Inform the user and say goodbye.
- 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 the Shipping Accounts Program

Hello, what is your username: EramoM

Hello eramom. Welcome back to your account.

Current shipping prices are as follows:

Shipping orders 0 to 100:	\$5.10 each
Shipping orders 100 to 500:	\$5.00 each
Shipping orders 500 to 1000:	\$4.95 each
Shipping orders over 1000:	\$4.80 each

How many items would you like to ship: 867

To ship 867 items it will cost you \$4291.65 at \$4.95 per item.

Would you like to place this order (y/n): y

Okay. Shipping your 867 items.

Example Output 2:

Welcome to the Shipping Accounts Program

Hello, what is your username: allenj

Hello allenj. Welcome back to your account.

Current shipping prices are as follows:

Shipping orders 0 to 100:	\$5.10 each
Shipping orders 100 to 500:	\$5.00 each
Shipping orders 500 to 1000:	\$4.95 each
Shipping orders over 1000:	\$4.80 each

How many items would you like to ship: 235

To ship 235 items it will cost you \$1175.0 at \$5.0 per item.

Would you like to place this order (y/n): n

Okay, no order is being placed at this time.

Example Output 3:

Welcome to the Shipping Accounts Program

Hello, what is your username: smtihJ

Sorry, you do not have an account with us. Goodbye.

Conditionals Challenge 17: Coin Toss App

Description:

You are responsible for writing a program that will simulate flipping a coin n number of times.

Your program will present the user an option to see the result of each individual flip. Your program will also inform the user any time the number of heads flipped is equal to the number of tails flipped. Upon completion of all flips, your program will provide a summary table that shows the number and percentage of each flip.

Step By Step Guide:

- Print a welcome message.
- Get user input for how many times the user would like to flip the coin.
- Get user input for whether they would like to see the result of each individual coin flip or not.
 - This response should be case insensitive.
 - Any response that starts with y should be taken as a yes.
- Create a variable to store the number of times heads is flipped and set it equal to zero.
- Create a variable to store the number of times tails is flipped and set it equal to zero.
- Flip a coin for the correct number of times.
- For each flip; do the following:
 - To simulate a coin flip we will randomly pick a number from two possible states.
 - The random generation of numbers is outside the scope of basic python.
In order to generate random numbers we will need to import a library of extra code.
 - Type import random as the first line of code in your program.
 - Create a coin flip using the random library to generate a random integer; 0 or 1.
 - Let one of these values imply that a heads was flipped while the other implies that tails was flipped.
 - Use an if/else statement to increase the number of heads or tails flipped depending on the outcome of the coin flip.
 - If the user indicated that they want to see the individual results:
 - Display what was flipped.
 - If the number of heads and tails are equal:
 - Inform the user each time this occurred.
- Calculate the percentage of heads flipped.
- Calculate the percent of tails flipped.
- Round these values to two decimal places.
- Print out the results of the total trials as formatted below.
- 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 the Coin Toss App

I will flip a coin a set number of times.

How many times would you like me to flip the coin: 97

Would you like to see the result of each flip (y/n): n

Flipping!!!

At 2 flips, the number of heads and tails were equal at 1 each.

At 8 flips, the number of heads and tails were equal at 4 each.

At 10 flips, the number of heads and tails were equal at 5 each.

At 12 flips, the number of heads and tails were equal at 6 each.

At 14 flips, the number of heads and tails were equal at 7 each.

At 20 flips, the number of heads and tails were equal at 10 each.

Results Of Flipping A Coin 97 Times:

Side	Count	Percentage
Heads	42/97	43.3%
Tails	55/97	56.7%

Example Output 2:

Welcome to the Coin Toss App

I will flip a coin a set number of times.

How many times would you like me to flip the coin: 10

Would you like to see the result of each flip (y/n): YEAH

Flipping!!!

TAILS

HEADS

At 2 flips, the number of heads and tails were equal at 1 each.

TAILS

TAILS

HEADS

HEADS

At 6 flips, the number of heads and tails were equal at 3 each.

TAILS

HEADS

At 8 flips, the number of heads and tails were equal at 4 each.

HEADS

TAILS

At 10 flips, the number of heads and tails were equal at 5 each.

Results Of Flipping A Coin 10 Times:

Side	Count	Percentage
Heads	5/10	50.0%
Tails	5/10	50.0%

Conditionals Challenge 18: Voter Registration App

Description:

You are responsible for writing a program that will simulate registering to vote. If a user is 18 or older, your program will present them with a list of potential political parties to register for. Upon choosing a party, your program will confirm that the user has registered and print a specific message depending on the party joined.

Step By Step Guide:

- Print a welcome message.
- Get user input for their name.
- Get user input for their age.
- Create the following list that holds the following political parties:
 - Republican, Democratic, Independent, Libertarian, Green]
- If the user is old enough to vote print a message informing the user they are old enough to vote.
 - List all the possible political parties to join as formatted below.
 - Get user input for the party they wish to join.
 - If the chosen party is in the list of voting parties print a message they have joined.
 - If the party is republican or democratic inform the user that it is a major party.
 - Elif the party is independent inform the user that they are an independent person.
 - Else inform the user that their party is not a major party.
 - Else inform the user that the party they chose is not a given party.
- Else inform the user they are not old enough to vote.
- 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 the Voter Registration App

Please enter your name: mike

Please enter your age: 16

You are not old enough to register to vote.

Example Output 2:

Welcome to the Voter Registration App

Please enter your name: mike

Please enter your age: 21

Congratulations Mike! You are old enough to register to vote.

Here is a list of political parties to join.

- Republican
- Democratic
- Independent
- Libertarian
- Green

What party would you like to join: republican

Congratulations Mike! You have joined the Republican party!

That is a major party!

Example Output 3:

Welcome to the Voter Registration App

Please enter your name: mike

Please enter your age: 33

Congratulations Mike! You are old enough to register to vote.

Here is a list of political parties to join.

- Republican
- Democratic
- Independent
- Libertarian
- Green

What party would you like to join: INDEPENDENT

Congratulations Mike! You have joined the Independent party!

You are an independent person!

Conditionals Challenge 19: Guess My Number App

Description:

You are responsible for writing a program that will play the classic “Hi Low” game. Your program will randomly pick a number between 1 and 20. Users will then guess the number. With each guess, your program will respond that the user’s guess is either too high or too low. When the user guesses correct, or after 5 guesses, your program will end the game and summarize the results.

Step By Step Guide:

- Print a welcome message.
- Get user input for their name.
- Inform the user that you are thinking of a number between 1 and 20.
- Use the random library to generate a random integer between 1 and 20.
 - Type import random as the first line of code in your program.
- Allow the user to guess 5 times. Each time do the following:
 - Get the users guess.
 - If the guess is too low inform them.
 - Elif the guess is too high inform them.
 - Else the guess is the number use the break command to exit the for loop.
 - Google or reference the python documentation on how the break command works.
- If the user got the number correct, inform them and tell them how many guesses it took.
- If the user did not get the number correct after the 5th guess, let them know they lost.
- 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 the Guess My Number App

Hello! What is your name: mike

Well Mike, I am thinking of a number between 1 and 20.

Take a guess: 15

Your guess is too low.

Take a guess: 20

Your guess is too high.

Take a guess: 19

Good job, Mike! You guessed my number in 3 guesses!

Example Output 2:

Welcome to the Guess My Number App

Hello! What is your name: MIKE

Well Mike, I am thinking of a number between 1 and 20.

Take a guess: 1

Your guess is too low.

Take a guess: 20

Your guess is too high.

Take a guess: 2

Your guess is too low.

Take a guess: 19

Your guess is too high.

Take a guess: 3

Your guess is too low.

Game Over. The number I was thinking of was 10.

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 :-(