Module 04

While Loops – Continued

Emily Zhao
CSCI–UA.0002

Agenda

- Poll Everywhere Review
- Continue practicing While Loops

Review:
pollev.com/emilyzhao

Do we need a while loop to solve this problem?

- Write a program that asks the user for two numbers
- If the first number is greater than the second number, add the numbers and display the total
- If the second number is greater than the first number, multiply the numbers and display the product

```
Enter 1st number: 5

Enter 2nd number: 10

Product: 50
```

Do we need a while loop to solve this problem?

- Write a program that asks the user for two numbers. Only accept positive values
- If the first number is greater than the second number, add the numbers and display the total
- If the second number is greater than the first number, multiply the numbers and display the product

```
Enter 1st number: 5

Enter 2nd number: -10
Sorry, try again
Enter 2nd number: 10

Product:  50
```

Do we need a while loop to solve this problem?

- Ask the user to enter in 5 price values
- Add these values to a total variable
- Print out the total at the end of the program plus 7% sales tax

```
Enter price: 1.00
Enter price: 2.00
Enter price: 3.00
Enter price: 3.00
Enter price: 1.00

Your total:  10.00
Tax:  0.70
Grand total:  10.07
```

Do we need a while loop to solve this problem?

- Ask the user to enter in a potentially ***unlimited*** number of price values
- Add these values to a total variable
- Print out the total at the end of the program plus 7% sales tax

```
Enter price, 0 to end: 1.00
Enter price, 0 to end: 2.00
Enter price, 0 to end: 3.00
Enter price, 0 to end: 3.00
Enter price, 0 to end: 1.00
Enter price, 0 to end: 0

Your total:  10.00
Tax:  0.70
Grand total:  10.07
```

Two ways to set up your while loops

while condition == True:

Keeps looping until the condition is no longer true (either arithmetic happens so that it is no longer true, or we set the condition to false)

while True:

Immediately enters and continues to loop indefinitely until the program encounters a break

How many times will the following loop run?

```
a = 0
b = 4

while a < b:
    a += 1
```

```
a = 0
b = 4

while a < b:
    a += 1  # Enters the loop: a = 0,
            # Run 1: a -> 1,
            # Run 2: a -> 2,
            # Run 3: a -> 3,
            # Run 4: a -> 4,
            # -> will not enter loop a 5th time
```

```
# set up control variable
keepAsking = True

if keepAsking == True: # enters the loop

    # will keep asking user to give them an integer
    num = int(input("Give me an integer: "))

    # until the condition is no longer true
    if num > 0:
        print("Thank you for inputting a valid integer")
        keepAsking = False
    else:
        print("Keep trying")
        # go back to the top of the loop and ask for input again
        # because keepAsking is still true
```

```
while True: # enter loop immediately

    # will keep asking user to give them an integer
    num = int(input("Give me an integer: "))

    # until the program reaches a BREAK
    if num > 0:
        print("Thank you for inputting a valid integer")
        break
    else:
        print("Keep trying")
        # go back to the top of the loop and ask for input again
```

Programming Challenge: Marbles

- Assume you have a jar that contains 5 marbles. The jar can hold 10 marbles total.
- Continually ask the user if they want to add or remove a marble
- If they add a marble you should increase the total # of marbles in the jar. If the jar is full tell the user and end the program.
- If they remove a marble you should decrease the # of marbles in the jar. If the jar is empty you should tell the user and end the program.

```
marbles = 5

while True:
    response = input("Would you like to add or remove a marble?: ")

    if response == "add":
        marbles +=1
    elif response == "remove":
        marbles -=1

    if marbles == 10:
        print("The jar is full.")
        break
    if marbles == 0:
        print("The jar is empty.")
        break
```

Programming Challenge: Grocery Shopping

Write a program that asks the user for an item's name and then its price

When the user enters the word "end" you should stop asking the user for item names and prices and display the following:

- The list of items they purchased
- Their total bill
- The cost of the highest priced item
- The cost of the lowest priced item

```
Enter an item's name (or type 'end'): apple
Enter a price for item: 2.00
Enter an item's name (or type 'end'): banana
Enter a price for item: 1.00
Enter an item's name (or type 'end'): peach
Enter a price for item: .5
Enter an item's name (or type 'end'): end
Your list: apple, banana, peach
Total cost: 3.5
Cost of most expensive: 2.0
Cost of least expensive: 0.5
```

```python
# accumulator variables
total_cost = 0
all_items = ""
number_of_items = 0
minimum = 0.00
maximum = 0.00

# create while loop that runs until sentinel is triggered
while True:
    item_name = input("Enter an item's name (or type 'end'): ") # ask for item name
    if item_name == "end":  # deal with my sentinel immediately
        break
    item_price = float(input("Enter a price for item: ")) # ask for price
    if number_of_items == 0: # set first item equal to both max and min
        maximum = item_price
        minimum = item_price
    if item_price > maximum: # check and update maximum
        maximum = item_price
    if item_price < minimum: # check and update minimum
        minimum = item_price
    number_of_items += 1 # increment number of items
    all_items += item_name + " "  # add item name with space after
    total_cost += item_price  # add price to total

print("Your list:", all_items)
print("Total cost:", total_cost)
print("Cost of most expensive:", maximum)
print("Cost of least expensive:", minimum)
```

```
# accumulator variables
total_cost = 0
all_items = ""
number_of_items = 0
minimum = 0.00
maximum = 0.00

# create while loop that runs until sentinel is triggered
while True:
    item_name = input("Enter an item's name (or type 'end'): ") # ask for item name
    if item_name == "end":  # deal with my sentinel immediately
        break
    item_price = float(input("Enter a price for item: ")) # ask for price
    if number_of_items == 0: # set first item equal to both max and min
        maximum = item_price
        minimum = item_price
        all_items = item_name # set first item in list to first item
    if item_price > maximum: # check and update maximum
        maximum = item_price
    if item_price < minimum: # check and update minimum
        minimum = item_price
    number_of_items += 1 # increment number of items
    if number_of_items > 1:
        all_items += ", " + item_name  # add item name to running list with comma first
    total_cost += item_price  # add price to total

print("Your list:", all_items)
print("Total cost:", total_cost)
print("Cost of most expensive:", maximum)
print("Cost of least expensive:", minimum)
```

Programming Challenge: Prime Number Checker

- Write a program that asks the user for an integer
- Test to see if the number is prime. A prime number is any number that is only divisible by 1 and itself.

Homework

- Self-Paced Learning Module #5 (due Mon)
- Quiz #5 (due Mon)
- Assignment #4 (due Mon)