Learning Objectives
Students will be able to:

Content:
- Explain the purpose of a predefined/built-in function
- Explain the functions: abs(), pow(), int() round(), random
- Explain the math library functions: floor() and ceil()
- Explain the use of the import statement
- Explain the purpose of a function argument

Process:
- Write code that uses predefined functions

Prior Knowledge
- Python concepts from Activities 1-10

FYI: Predefined/built-in functions: segments of code already included in Python. print(), round(), abs(), pow(), int() are examples.

Arguments: The information that a function needs to work. Arguments are sent to the function between the parentheses ().

To use a function, call the function. input("Enter your name") is a call to the input function sending the string “Enter your name” as an argument.

Critical Thinking Questions:
1. Circle calls to predefined functions in the following program.

```
# Get input
name = input("Enter your name: ")

# Print results
print("Name: ", name)
```

2. Enter and execute the Python program on the right.
   a. What is the output for each statement?
      - print(abs(-4.67)) ________________
      - print(pow(5,3)) ________________
      - print(pow(49,.5)) ________________
      - print(int(34.8)) ________________
      - print(round(6.9)) ________________
      - import random
        print(random.randint(1,100)) ________________

   b. What is the difference between the round() function and the int() function?
3. Each of the following function calls return a value. What is the value returned for each line of code? Verify your answers by executing the code.
   a. \texttt{abs(4.5)}
   b. \texttt{int("678")}
   c. \texttt{round(-5.6)}
   d. \texttt{import random}
      \texttt{random.randint(4,10)}

   What is the purpose of “import random”? What happens if you omit that line of code?

4. Circle the \textbf{argument} in the call to the built-in function:
   \begin{verbatim}
   number = 45.78
   answer = round(number)
   \end{verbatim}

5. \texttt{answer = pow(4,3)}. What is/are the argument(s) in this code?

6. If a function contains more than one argument, do you think the order of the arguments makes a difference? Explain your answer with an example.

7. Execute the following lines of code:
   \begin{verbatim}
   import math
   x = 4.7
   y = 5.3
   z = -4.8
   a = -3.2
   print(math.ceil(x))
   print(math.ceil(y))
   print(math.ceil(z))
   print(math.ceil(a))
   print(math.floor(x))
   print(math.floor(y))
   print(math.floor(z))
   print(math.floor(a))
   \end{verbatim}

   a. Explain the purpose of the \texttt{ceil()} function.
   b. Explain the purpose of the \texttt{floor()} function.
   c. Why are the calls to the \texttt{floor()} and \texttt{ceil()} functions preceded by “\texttt{math.”}?
8. Enter and execute the following code:

```python
doAgain = True
while doAgain:
    num1 = int(input("Enter first number: "))
    num2 = int(input("Enter second number: "))
    num3 = int(input("Enter third number: "))
    num4 = int(input("Enter fourth number: "))
    maxNum1 = max(num1, num2, num3, num4)
    print("The largest of the four numbers is: ", maxNum1)
    another = input("Type 'y' to find another max number or any other key to quit.")
    if another != 'y':
        doAgain = False
        print("Done!")
```

a. What type of variable is ‘doAgain’? 

b. What does the program do? 

c. What does the following line of code do?
   ```python
   maxNum1 = max(num1, num2, num3, num4)
   ```

d. Experiment with the arguments in the `max()` function in the program to determine if the function must have four arguments. Provide an example for your answer.

e. What does the following code in the last two lines of the program do?
   ```python
   if another != 'y':
       doAgain = False
   ```

Application Questions: Use the Python Interpreter to check your work

1. Write a line of code that prints the integer portion of the number 21.45.

2. Write code that prompts the user for a floating point number and prints the smallest integer that is larger than the number the user entered.

3. Write the code to print a random number between one and six.
4. Assume that a user enters any number and that the number is stored in the variable `userNumber`. Write a line of code that converts the input to an integer. Then write a line of code that prints the positive value of the user’s input.

5. Write a line of code that calculates the square root of 900 and stores the result in the variable `answer`. 
