Happy Lunar New Year!

Chúc Mừng Năm Mới

新年快乐 / 新年快樂
Xīnnián kuàile

过年好 / 過年好
Guò nián hǎo

새해 복 많이 받으세요
saehae bog manh-i bad-euseyo

感谢互联网提供的图片!

Thank you Internet for the pic!
I just got fired from my job at the keyboard factory.

They told me I wasn't putting in enough shifts.


Thank you Steve!
Last Lecture

- Introduced **functions**

- Restructured our **OddEvenEncryption** program by adding two **functions**:
  - `encrypt(...)` and `decrypt(...)`

- We looked at another iterative statement:
  - `while` loop

- Lasted **OddEvenEncryption** program now posted on our course website!
Today’s Menu

• Investigating different kinds of functions
• Looking at the execution flow when Python executes our functions
Different kinds of functions?

- **Problem Statement:**
  - Create a function that prints "*" 5 times in a row
Different kinds of functions?

- **Problem Statement:**
  - Create a function that returns the sum of the first 10 integral numbers
Different kinds of functions?

- **Problem Statement:**
  - Create a function that asks the user for her/his name and prints the given greeting

- **Requirement:**
  - We must incorporate the user’s name within the greeting
Different kinds of functions?

- **Problem Statement:**
  - Create a function that returns the largest number out of 3 numbers

- **Requirement:**
  - We cannot use the \texttt{max()} built-in function
  - We cannot use the word \texttt{max} to name your function
Different kinds of functions

**Activity:** Fill this table using functions we have used so far!

<table>
<thead>
<tr>
<th>Function that returns a result (i.e., a returned value)</th>
<th>Function that takes argument(s)</th>
<th>Function that does not take argument(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function that returns a result (i.e., a returned value)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Function that does not return a result (i.e., a returned value)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Execution Flow with Functions

• Let’s first examine what happens to the execution flow when we call functions
  • using the Python code visualizer
Guidelines to follow!

• When creating a function, ask yourself these 3 questions:

1. What does the function do?
   • Tip: The answer should be used to name the function

2. Does the function needs data to operate i.e., data from the other functions or the main part of our program?
   • If so, the function must have parameters and the function must be called with arguments

3. Must the function return a value?
   • If the function produces a result then it must return it to the caller (not same as printing the result!)
Your turn!

- **Problem Statement:**
  - Create a function that prints a character x number of times in a row

- **Requirement:**
  - Let’s not forget about the *Generalization Principle*
Next lecture

- We shall introduce another field of Computing Science:
  - **Graphics and Animation**
- We shall introduce a fun Python module:
  - **Turtle**