

C++ Day 8 Commands, Functions, and Tricks:

void function:

- Purpose: creates a function that has no output value
- Note: void functions are generally used to *perform some action* instead of doing some kind of calculation or determination
- Examples:
 - Declare a new function called "CatchTheChicken ()" that has no output and has an `int`-type input called "numHands" and a `bool`-type input called "ShouldIDrop"

```
void CatchTheChicken(int numHands, bool ShouldIDrop);
```

create a string:

- Purpose: a string is a type of data, much like `int` or `float`. This datatype stores a set of characters—a piece of text—that can be treated like a single variable
- Note: to use strings, you must `#include <string>`
- Examples:
 - Create a string called "text", set its value to "Northeastern University" and then print it to the screen

```
#include <string>
...
string text="Northeastern University";
cout << text;
```

get individual characters from a string:

- Purpose: to retrieve a specific character from a string, place the number of the character desired in square brackets after the name of the string
- Note: the numbering of characters begins **at zero**
- Examples:
 - Print to the screen the 1st, 2nd, and 3rd character from the string "text" created in the previous example

```
cout << text[0] << text[1] << text[2];
```

- Set an `int` variable "x" equal to 5 and then print to the screen the xth character from the string "text"

```
int x = 5;
cout << text[x];
```

string operators:

- Purpose: a few operators work with string variables to modify or compare them
- Examples:
 - Create 2 strings, 1 called "first" and 1 called "second". Then set their values to "North" and "eastern" and print to the screen the concatenated (combined) string of both.

```
#include <string>
...
string first, second;
first = "North";
second = "eastern";
cout << first + second;
```

- Compare the variable created in the previous example called "second" to see if it equals "western". If so, correct the horrible mistake.

```
if(second == "western")
{
    second = "eastern";
}
```

- Create a string variable called "name" and then set its value to the user's input.

```
string name;
cin >> name;
```

string member functions:

- Purpose: perform various useful functions on or with strings
- Examples (all use the string called "text" that was created and initialized in the example above):

- determine the number of characters in the string.

```
text.size()
```

- determine if the string is empty (i.e., contains no characters).

```
text.empty()
```

- get the substring that starts at the 3rd character and continues for 4 characters from there.

```
text.substr(2,4)    // the 1st character is number 0,
                   // so the 3rd character is number 2
```