Basic Syntax (Python)

Chantilly Robotics (Team 612)

Hello World

- We’re doing the same thing everyone else is: a program that prints “Hello World”
- The purpose is to demonstrate basic output functionality and as an introduction to basic syntax
- This is a well known programming tradition, you ***must*** follow it

:P

Others are using http://repl.it
But we’re going to get started with a standalone Python interpreter and tool package that I’ve created.

Go to: <GOOGLE DRIVE LINK HERE> and download. See README.md for more information.

Example program

This is the first thing you see on the website http://cpp.sh

- Delete all of this

```cpp
// Example program
#include <iostream>
#include <string>

int main()
{
  std::string name;
  std::cout << "What is your name? ";
  getline (std::cin, name);
  std::cout << "Hello, " << name << "!\n";
}
```

The Hello World program

- Replace the previous text with the program to the right
- This is case sensitive

```
#include <iostream>

int main() {
     std::cout << "Hello World!" << std::endl;
}
```

Let’s break it down!

Comments

- Text in source code that is ignored by the compiler
- Used to explain, debug, and documentation

Comment usage

- Text in green are comments
- Two types of comments
  - Single line
    - Denoted by //
  - Multiple line
    - Starts with /*
    - Ends with */

```
#include <iostream>

int main() {
     /*
      * Multi-line
      * Comment
      * Here
      */
     std::cout << "Hello World!"; //Prints hello world
     //single line comment here
}
```

Whitespace

- Whitespace is ignored by the compiler*
- This includes:
  - Spaces
  - Indents
  - Newline characters

* Notable exceptions are preprocessor directives (we’ll cover this later) and string literals

These two are the same (to the compiler)

```
#include <iostream>

int main() {
    std::cout << "Hello World!";
}
```

```
#include <iostream>
int
main()
{std::cout
<< "Hello World!"
;}
```

Proper use of whitespace

- Whitespace should be used to organize code
- Whitespace should be used to split up lines that are very long
- Organization should be consistent
- Whitespace should produce readable code

One is more readable (to human eyes)

```
#include <iostream>

int main() {
    std::cout << "Hello World!";
}
```

Superior

```
#include <iostream>
int
main()
{std::cout
<< "Hello World!"
;}
```

How does the compiler separate statements?

- The semicolon character `;` is used to end statements
- This is why whitespace can be safely ignored in most places

Example program

- Notice all the highlighted semicolons

```
// Example program
#include <iostream>
#include <string>

int main() {
  std::string name;
  std::cout << "What is your name? ";
  getline (std::cin, name);
  std::cout << "Hello, " << name << "!\n";
}
```

Our code

```
std::cout << "Hello World" << std::endl;
```

- `std::cout` - An output stream (more on this way later) that takes in characters and prints them on the screen
- `<<` - Stream insertion operator (more on this way, WAY later)
- `"Hello World"` - The string we want to print
- `std::endl` - Tells the output stream to enter a line break into the screen
- `;` - Tells the compiler this is the end of the line