What Data Do We Have?
Challenge Solution

//********************************************************
// File name: hello.cpp
// Author:    Chadd Williams
// Date:      09/01/2006
// Purpose:   This program asks for a users name and then
displays a personalized message.
//********************************************************

#include <iostream>
#include "stdafx.h"
#include <string>
using namespace std;

int main()
{
  string name;
  cout << "What is your name? " << endl;
  cin >> name;
  cout << "Hello " << name << "!" << endl;
  return 0;
}
Today

On Wednesday I showed you a C++ program that displays a personalized message to the user.

What are the main components of that program?

Today we will

- learn how C++ stores data
- Some of the different types of data that C++ can store
main Function

- `int main()`
- Marks the beginning of a function
- A function is a group of one or more programming statements
- The set of parentheses indicate a function
- C++ is case-sensitive
  - `int Main()` is incorrect!!!
**cout Object**

- `cout` object is the standard output object
- The monitor is the standard output device
- `cout` is a stream object and works with streams of data
  - Streams of characters
cout Object

- Output operator (insertion operator): `<<`
- Standard output (monitor screen): `cout`
- The value to the right of the operator (right operand) is displayed on the screen
  - If the right operand is within double quotes, then it is output exactly as it appears
    - The exception is if it is an escape character `\`
  - If the right operand is a variable or constant, then the value of that variable or constant is output
cout Object

- What is the output?
  
  ```
  cout << "Enter the distance in miles" << endl;
  cout << "The distance in kilometers is " << kms << endl;
  ```

- You must always use the insertion operator `<<` to separate the different components you wish to output

- `endl` will move the cursor to a new line

- All output statements must end in a semicolon

- Output strings within double quotes "" should always appear on one line
cout Object

- `<<` is used to separate the different output items

- Example:
  - `cout << "Type your name, then press enter" << endl;`

- It is illegal to break up the string literals across lines
  - `cout << "Type your name, then press enter" << endl;`
  - `Is illegal!!`
cout Object

- Other ways of outputting the same message

```cpp
cout << "Type your name, " << "then press enter" << endl;
```

```cpp
cout << "Type your name, ";
```

```cpp
cout << "then press enter" << endl;
```

- Everything will output to the same line unless you specify otherwise
Problem

- What is the output?

```cpp
cout << "My name is: ";
cout << "Doe, Jane." << endl;
cout << "I live in ";
cout << "Ann Arbor, MI ";
cout << "and my zip code is "
    << 48109 << ". " << endl;
```
Problem

- Write the C++ statements necessary to perform the following operation
  - Display the message below onto the screen
    "C++ is a useful language to know"
Escape Characters

- These are special characters that can be output.
- They are always preceded by a backslash \.
- Examples of escape characters include:
  - \n: moves the cursor to the beginning of the next line. Equivalent to endl.
  - \t: moves the cursor to the next tab stop.
  - \": outputs the double quotes.
  - \\: displays the backslash.
  - \": outputs the double quotes.
  - /square4: equivalent to endl.
Examples

- What is the output?
  - `cout << "This is a C++ program\n";`
  - `cout << "This is a \nC++ program";`
  - `cout << "\"This is a C++ program\"";`
  - `cout << "This is a\tC++\tprogram";`
Variables

- A variable is a named storage location for holding data.

- Part of the job of programming is to determine how many variables a program will need.

- Let’s look at program 2-7 on p. 41, also on the next slide.
Variables

```cpp
#include "stdafx.h"
#include <iostream>

using namespace std;

int main()
{
    int number;

    number = 5;
    cout << "The value of number is " << number << endl;
    cout << "The value of number is " << number << endl;

    number = 7;
    cout << "Now the value of number is " << number << endl;

    return 0;
}
```
Variable Definition

- `int number;`

- Tells the compiler
  - The variable’s type (int)
  - The variable’s name (number)

- `int` is short of integer

- Variable definitions end with a semicolon
Assignment

- `number = 5;`
- `=` is an operator that copies the value on its right into the variable on its left
- The item to the left of the `=` operator must be a variable
Program Output

- What do you think is the program’s output?
String Literals

• Placing quotations around a variable name changes it to a string literal

```
cout << "The value of number is " << number << endl;
```

• Changes it to a string literal

What is the output of the statement

```
   cout << "endl";
```

String Literals
Exercises

Which of the following are legal C++ statements?

- $a = 7$
- $7 = a$
- $7 = 7$
• In today's lecture we covered:
  - `main` function
  - `cout` object
  - How data that is used by a program can be declared and stored
  - In today's lecture we covered p. 31 - 45 of your textbook