

CSCI 1370

FEBRUARY 6, 2016

REVIEW

Programs are divided up into **functions**.

```
#include <iostream>
using namespace std;

int main()
{
    cout << "Hello world!";
}
```

A **function** is just a container. For example:

```
#include <iostream>
using namespace std;
```

```
int main()
{
    cout << "Hello world!";
}
```

The **main** function is special (execution starts with main).

```
#include <iostream>  
using namespace std;
```

```
int main()  
{  
    cout << "Hello world!";  
}
```

PROGRAM EXECUTION

```
#include <iostream>  
using namespace std;
```

```
int main()  
{  
    cout << "Hello world!";  
}
```

In general, **functions** are used for code organization and reusability.

EXAMPLE: A PROGRAM WITH TWO FUNCTIONS

```
#include <iostream>
using namespace std;
```

```
void sayGoodbye()
{
    cout << "Goodbye!";
}
```

```
int main()
{
    cout << "Hello world!";
}
```


PROGRAM EXECUTION

```
#include <iostream>
using namespace std;
```

```
void sayGoodbye()
{
    cout << "Goodbye!";
}
```

```
int main()
```

```
{
```

```
    cout << "Hello world!";
```

```
}
```

USING FUNCTIONS

To use a function, we must make a **function call**.

To *call* our function (named “sayGoodbye”) we need to add the following statement to our code:

```
sayGoodbye( );
```

EXAMPLE: A PROGRAM WITH TWO FUNCTIONS

```
#include <iostream>
using namespace std;

void sayGoodbye()
{
    cout << "Goodbye!";
}

int main()
{
    cout << "Hello world!";
}
```

EXAMPLE: A PROGRAM WITH TWO FUNCTIONS

```
#include <iostream>
using namespace std;

void sayGoodbye()
{
    cout << "Goodbye!";
}

int main()
{
    cout << "Hello world!";
    sayGoodbye();
}
```

PROGRAM EXECUTION

```
#include <iostream>  
using namespace std;
```

```
void sayGoodbye()
```

```
{
```

```
    cout << "Goodbye!";
```

```
}
```

```
int main()
```

```
{
```

```
    cout << "Hello world!";
```

```
    sayGoodbye();
```

```
}
```

EXERCISE

With a partner, write statements in the `main` function that call `anotherFunction` and complete the output to get “Hello world!” printed to the console.

```
void sayWorld()
{
    cout << "world";
}

int main()
{
    // Your statements here.
}
```

EXERCISE

With a partner, now *add a third* function (call it `functionC`), which prints out “glorious”. Modify your code in `main` such that the phrase, “Hello glorious world!” is printed in the console.

```
void sayWorld()
{
    cout << “world”;
}

int main()
{
    cout << “Hello “;
    sayWorld();
    cout << “!”;
}
```

FUNCTIONS, CONT.

WRITING FUNCTIONS

- To write a function, you must specify a **header** and a **body**.
- The **header** has three parts:
 - *a return type* (e.g., `int`),
 - *a name* (e.g., `main`), and
 - *a list of parameters* between parentheses.
- The **body** is a sequence of statements placed between two curly brackets (`{}`).

THE `main` FUNCTION

```
int main()  
{  
    //statements to execute  
}
```

WRITING FUNCTIONS

- To write a function, you must specify a **header** and a **body**.
- The **header** has three parts:
 - a *return type* (e.g., `int`, `void`),
 - a name (e.g., `main`, `sayGoodbye`), and
 - a *list of parameters* between parentheses.
- The **body** is a sequence of statements placed between two curly brackets (`{}`).

FUNCTION: MAIN

```
int main()  
{  
    //statements to execute  
}
```

FUNCTION: MAIN

```
int main()  
{  
    //statements to execute  
}
```

FUNCTION: MAIN

```
int main()  
{  
    //statements to execute  
}
```

WRITING FUNCTIONS

- To write a function, you must specify a **header** and a **body**.
- The **header** has three parts:
 - *a return type* (e.g., `int`, `void`),
 - *a name* (e.g., `main`, `sayGoodbye`), and
 - *a list of parameters* between parentheses.
- The **body** is a sequence of statements placed between two curly brackets (`{}`).

FUNCTION: MAIN

```
int main()  
{  
    //statements to execute  
}
```


FUNCTION: sayGoodbye

```
void sayGoodbye()  
{  
    cout << "goodbye";  
}
```

FUNCTION: sayGoodbye

```
void sayGoodbye()  
{  
    cout << "goodbye";  
}
```

FUNCTION: sayGoodbye

```
void sayGoodbye()  
{  
    cout << "goodbye";  
}
```

FUNCTION: sayGoodbye

```
void sayGoodbye()  
{  
    cout << "goodbye";  
}
```

FUNCTION: sayGoodbye

```
void sayGoodbye()  
{  
    cout << "goodbye";  
}
```

FUNCTION: SUM

```
int sumOf(int a, int b)
{
    return a + b;
}
```

FUNCTION: SUM

```
int sumOf(int a, int b)
{
    return a + b;
}
```

FUNCTION: SUM

```
int sumOf(int a, int b)
{
    return a + b;
}
```


FUNCTION: SUM

```
int sumOf(int a, int b)
{
    return a + b;
}
```

FUNCTION: SUM

```
int sumOf(int a, int b)
{
    return a + b;
}
```

EXAMPLE: USING FUNCTIONS

```
#include<iostream>;  
using namespace std;  
  
int sumOf(int a, int b)  
{  
    return a + b;  
}  
  
int main()  
{  
    int result = sumOf(5, 10);  
    cout << result;  
}
```