C Output (Print Text)

Output (Print Text)

To output values or print text in C, you can use the `printf()` function:

Example

```
#include <stdio.h>

int main() {
  printf("Hello World!");
  return 0;
}
```

Double Quotes

When you are working with text, it must be wrapped inside double quotations marks `""`.

If you forget the double quotes, an error occurs:

Example

```
printf("This sentence will work!");


printf(This sentence will produce an error.);
```

Many `printf` Functions

You can use as many `printf()` functions as you want. **However**, note that it does not insert a new line at the end of the output:

Example

```
#include <stdio.h>

int main() {
  printf("Hello World!");
  printf("I am learning C.");
  printf("And it is awesome!");
  return 0;
}
```

New Lines

To insert a new line, you can use the `\n` character:

Example

```
#include <stdio.h>

int main() {
  printf("Hello World!\n");
  printf("I am learning C.");
  return 0;
}
```

You can also output multiple lines with a single `printf()` function. However, this could make the code harder to read:

Example

```
#include <stdio.h>

int main() {
  printf("Hello World!\nI am learning C.\nAnd it is awesome!");
  return 0;
}
```

Tip: Two `\n` characters after each other will create a blank line:

Example

```
#include <stdio.h>

int main() {
  printf("Hello World!\n\n");
  printf("I am learning C.");
  return 0;
}
```

What is \n exactly?

The newline character (`\n`) is called an **escape sequence**, and it forces the cursor to change its position to the beginning of the next line on the screen. This results in a new line.

Examples of other valid escape sequences are:

Escape Sequence	Description
\t	Creates a horizontal tab
\\	Inserts a backslash character (\)
\"	Inserts a double quote character