Type Conversion

This lesson discusses the conversion variable types to other inbuilt types using example

WE'LL COVER THE FOLLOWING ^Converting ValuesExample

Converting Values

In the previous lesson, we covered the various types a variable in Go can take. Now, we will discuss how the type of a given variable can be changed. Converting values from one type to another is fairly simple in Go. The expression T(v) converts the value v to the type T.

Example

Given below are some example numeric conversions:

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

int main() {
   // your code goes here
   cout << "Hello World";
   return 0;
}</pre>
```

6

These can be put more simply:

var f float64 = float64(i)

var u uint = uint(f)

```
i := 42
f := float64(i)
u := uint(f)
```

Go assignment between items of different type requires an explicit conversion which means that you manually need to convert types if you are passing a variable to a function expecting another type. Now that we know how variable types can be changed, in the following section, we will look into how you can convert a given value to another specific type.