Conversion between char arrays, strings, and other types

Text may be stored in a character array or string, and we may want to convert to another type.

Eg: "400" to 400

.- or go the other way...

Eg 400 to "400"

... and we may want to split a string into different parts

Eg "Heat oven to 400° and bake for 15 minutes"

"900 Fifth Street, Nanaimo, BC V9R 5S5"

⟨cstdlib⟩

atoi/f   ascii to int

```
int i = atoi("106");    // i now 106
float f = atof("3007.95")   // f now 3007.95
                ↑
```

Works for literals and char arrays

⟨string⟩   has   stoi   and   stof

```
string s = "409";
int i = stoi(s);

string s2 = "39.5";
float f = stof(s2);
```

```cpp
#include <sstream>
#include <string>
#include <iostream>

using namespace std;

:
string text = "900 Fifth Street";
istringstream stream(text);   // create stream
                              // variable from
                              // text
int x;
stream >> x;   // x is now 900

// Can use getline, just as with cin

istringstream stream2("900 Fifth Street\nNanaimo");
string line1;
getline(stream2, line1, '\n');   // Reads
                                 // "900 Fifth Street"
                                 // into line1.
```

... still using <sstream>

Can use ostringstream to write text
into a stream, then use the stream's .str()
to get it into a string

```cpp
int x = 10;
ostringstream strm;
strm << "x = " << x << endl;
string s = strm.str();
// s now is "x = 10\n"
```

Converting String to char

```
char text[6] = "abcde";   // need space
                          // for \n
                          // so 6 chars
```

```
String s = text;
```

To get content of string as an array
we need .c_str()

```
strcpy(text, s.c_str());
// gets content of s as a null-terminated
// char array — thats the type that strcpy
// needs.
```