Class Data Members

This lesson will go into the details of the data members of a class.

We've learned that the data members contain all the information we store in a class. All the data members have to be defined at compile time.

It is a very safe practice to make our member variables *private*. Making them *public* could possibly crash the application because any external force could manipulate them in any way.

Data Types of Member Variables

C++ gives us a lot of freedom in selecting the data type of a data member. We can choose any of the in-built types such as `int`, `double` etc. Arrays, vectors, and pointers can also be used. The object of our custom class could have a number of arrays and variables! pretty cool right?

Here's an example of a `Student` class and its data members:

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

class myClass {
  // All private members
  string name;
  int age;
  string *address;
  char grades [10]; // A student can have a maximum of 10 grades
};
```

We should always be careful with arrays inside classes. We should have

appropriate checks in place to make sure our program never goes out of

bounds due to an array. A good practice is to store the size of the array in a variable. This way, we'll always remember the maximum capacity.

Objects of Other Classes as Data Members

This is another feature which adds to the flexibility of classes. We can use an object from our own class as a data member in our other classes. This is a slightly more advanced concept which we will study later on in the course.

All the data members we have seen till now are simply declared variables. They don't have any meaningful values. Be patient, we will learn how to assign values to our private data members through **member functions**.

We will discuss member functions in the next lesson.