Summary

- Encapsulation : variables + functions
- Data hiding : variables with private access
- this keyword
- const keyword usage
- constructor
- types of constructor - default, parameterized, copy
- synthesized operations supplied by compiler

- initializer list with constructors
- defining functions outside the class
- splitting code into headers + source files

```
int x;
std::cout << x << "\n";
x=10;

int y=20;
std::cout << y << "\n";
```

Examples

- Box
- Account
- Point (6.5)
- MyTime (6.9)

Constructor
- same name as class name
- no return type
- initialize object state (each member, initial value)
- implicitly called just after object creation (scope rules)

Constructor overloading
Parameterized Constructor

Default Constructor -- which takes no arguments
If we don't implement any constructor, compiler provides one
do-nothing default constructor (Which leave members with default values)
You may call it as synthesized default constructors

If any constructor is implemented by user, compiler don't provide
synthesized default constructor

If no user implemented copy ctor, compiler provides a synthesized
copy ctor, which will copy member values from exising to new object
(But this is shallow copy, which will fail for non-trivial classes)

Multi files

box.h
box.cpp
main.cpp

g++ main.cpp -c # main.o, compile only
g++ box.cpp -c # box.o
g++ box.o main.o -o boxdemo, linking

Further topics:-
* class diagrams (plantuml)
* need for destructor, MyStack example
* MyString example (Non trivial)
* static members
* friend class

C++ Concepts
- Classes & Objects
  (Constructors, Destructors)
- Operator Overloading
- Inheritance & Virtual functions
- Templates
other :
- File Handling
- Exception handling