Agenda

- Ctor & Types
- Destructor
- Mutators
- Inspectors
- Constant
- Ctor Members Initializer list
- Dynamic Memory Allocation

Member functions

- 1. Constructor
- 2. Destructor
- 3. Mutators
- 4. Inspectors
- 5. Faciliatators

Constructor (demo01.cpp)

- It is a special member function of the class
- why it is special
  - Its name is same as that of class name
  - It does noy have any return type
  - It gets called automatically whenever the object is created
- Ctor is used to initialize your object with the user seecified default values
- Ctor gets called only once in the life time of an object
- Types of Ctor
  - 1. Deafult Ctor / Parameterless Ctor
  - 2. Parametrized Ctor
  - 3. Copy Ctor

Destructor (demo02.cpp & demo03.cpp)

- It is a special member function of the class
- why it is special
  - Its name is same as that of class name with tild(~) sign
  - It does noy have any return type
  - It gets called automatically whenever the object goes out of scope
- Dtor calling sequence is exactly opposite to that of Ctor calling sequence

Mutators (demo04.cpp)

- The functions which accept single argument and chages/mutates the value of a single data member of the class, are called as mutators
- Mutatators are also called as Setters

Inspectors (demo05.cpp)

- The functios which return the value of single datamember of the class as it is are called as Inspectors
- Inspectors are also called as Getters

Constant Data Member(demo06.cpp)

- cont data members must be initialized inside Ctor member initializer list
- Once initialized you cannot change its value

Constant Member Function (demo07.cpp)

- If inside the functions you are jjust uisng the values of datamemebrs of the class and not modifying it then such member functions can be made as const

Constant Object (demo08.cpp)

- If you make the object as constant then you can only call const functions on that object.
- You cannot call non constat functions on const objects

Dynamic Memory Allocation (demo09.cpp)