# Agenda

```
Shallow Copy
Deep Copy
Friend Function
Static
Template
```

# Shallow Copy (demo.cpp)

```
- If you class have normal data members then in such case if you try to copy one
object into another at the time of object creation then your default copy ctor
does the shallow copy
- Shallow copy works fine for normal data memebers
```

# Deep copy (demo01.cpp)

```
- if you class consists of pointer type of data members
- and if you allocate the memory dynamically
- then while assigning the already created object to the newly created object it
will again does the shallow copy.
- in this shallow copy both the objcts will point to the same dynamically
allocated memory.
- in such case a change in one object will cause change in another object as well.
- to avoid this we should do deep copy.
- to to do the deep copy write your own copy ctor.
```

# Friend Function (demo02.cpp)

```
- If you want to access the private/protected members of a class in your global
functions directly on the object uisng dot operator then in such cases declare
that gloabl functions as a friend function inside that class
```

# Static (demo03.cpp)

```
1. Data member
    - It will be shared with all the objects of the class
    - It must be initialized outside the class using classname and scope
resolution operator
```

```
2. Member function
    - These function can be called using object as well as using class name and
scope resolution operator
    - these functions do not get this pointer
    - these functions can only access static data member inside them.
```

## Template

```
1. Function Template (demo04.cpp)

2. Class Template (demo05.cpp)
```

## Home work

```
-Friend function
-class datamemebrs & member functions
-this pointer
- arithmetic operators (+,-,*,/)
- differnt types of operands.
- difference between error logic and business logic ?
```