Agenda

- Friend Function
- Operator Overloading
- Template
- Exception Handling

Friend Function (demo01)

- It is a non member function of a class.
- All global functions that we declare are considered as non member functions
- If we declare a non member function as friend inside a class then such functions are able to access all the private members of that class directly on class object.
- This pointer is not passed to the friend functions
- If we are accessing the private members of the class multiple number of times in the non member functions with help of inspectors and mutators then instead of using those functions multiple times declare the function as a friend inside the class.

Operator Overloading (demo02 and demo03)

- operators can work for fundamental types of operands.
- however they cannot work for the derived/user defined types of operands
- to make this operator work for the user defined types we need to overload them.
- to overload the operator we need to define an operator function.
- we can overload the operators in two ways

1. using member function
2. non member function

- We cannot overload below functions as non member function, they must be overloaded as a member function only

1. Assignment operator `=`
2. function call operator `()`
3. subscript operator `[]`
4. Arrow operator `->`

- Their are some operators that you cannot overload

Template (demo04 and demo05)

- If you want to write generic programs in cpp we should use template
- We can have two types of template

1. function template
2. class template

Exception Handling

- Exception handling is used to seperate the business logic from the error handling logic
- we use below keywords to perform exception handling

1. try
2. catch
3. throw