# Agenda

```
- Access Specifier in structure
- Namespace
- scope resolution operator
- cin,cout
- Inline function
- Function Overloading
```

# Access Specifiers in structure (demo01.cpp)

```
- by default members of structure are public.
1. private
    - if we dont want to allow the visibility of members outside the structure
then make them as private
2. public
 - if we want to allow the visibility of members outside the structure then make
them as public
```

# Namespace (demo02.cpp to demo07.cpp)

```
- Their are "n" no of variations for examples in namespace.
- It is a container which is used to hold varaibles/functions/structures/ classes
- To access the members of a namespace you have to use one operator.
- the operator that we are going to use is ( :: ) scope resolution operator
- you cannot initiatate the namespace
```

# Cin Cout (demo08.cpp & demo09.cpp)

```
- to take input from user and display the output on console
- it is replacemnet for your printf() and scanf()
- cin is an external object of istream class
- cout is an external object of ostream class
- these objects are kept inside the std namespace
- cin -> (>>) Extraction opertaor
- cout -> (<<) Insertion Operator
```

# Inline Function (demo10.cpp)

```
- It is just a request made to the compiler
- It is accepted if the execution time is very less as compared to the function
call.
- For inline function the compilation time increases but teh execution time
decreases.
```

## Function overloading (demo11.cpp)

```
- having multiple functions with same name but different signature
- For function overloading younshould follow the rules as per below
1. Keep the name of functions same
2. The signature should be different
    a. if type of arguments and count are same then no of parameters should be
differnt for the next function
    b. if count of parameters is same then their types should be different
    c. If type and no of parameters are same then their order should be differnet
3.Return type is not considered

- Function overloading is an example of Polymorphism
- It is an example of compile time polymmorphism
- It is possible beacuse of a concept called as name mangaling
```