

`/*-- Note: I have done FILE_NAME changed edited on 25-3-2022-----*/`

Access Modifier

Private -Public & Public-Private

```
#include <iostream>
using namespace std;
class Sample
{
    private:
        void read(); // Private Member Function

    public:
        void display(); // Public Member Function
};

void Sample::read()
{
    cout<<"I am private class";
}

void Sample::display()
{
    cout<<"I am a public class";
}

int main()
```

```

{
    Sample S;
    S.display();
    S.read(); // Generates Error as we are calling Private Member Function
    return (0)
}

```

Explanation:-

Generates Error:-

```

main.cpp:36:12: error: 'void Sample::read()' is private within this context
   36 |     S.read();
      |           ^
main.cpp:22:6: note: declared private here
   22 | void Sample::read()
      |     ^~~~~~

```

Solution:-

Solution 1:-

Private member Function should be called from Public Member Function.

For Example:-

```

void Sample::read() //Private Member Function
{
    cout<<"I am private class"<<endl;
}

```

```

void Sample::display() // Public Member Function
{
    read(); // Calling Private Member function from Public Member Function
    cout<<"I am a public class"<<endl;
}

```

Output:-

I am private class

I am a public class

Solution 2:- Using Friend Function

```

#include <iostream>
using namespace std;
class Member
{
    private:
        int private_mem=0; // private member

    public:
        friend int func (Member); //friend Function
};

int func (Member m)
{
    m.private_mem=5; // Friend Function Accessing Private Member
    return m.private_mem;
}

```

```

int main()
{
    Member m;

    cout<< "Friend Function changed value of Private Member::"<<func(m);

    return(0);
}

```

Output of the following Program:-

Friend Function changed value of Private Member from 0 to :: 5

2)Protected -Public&Public-Protected

```

#include <iostream>

using namespace std;

class Sample
{
    protected:

    void read();//Protected Member Function

    public:

    void display();//Public Member Function
};

void Sample::read()
{
    cout<<"I am protected class"<<endl;
}

```

```

}

void Sample::display()
{
    cout<<"I am a public class"<<endl;
}

int main()
{
    Sample S;

    S.display();

    S.read();// Generates Error as calling Protected Member Function

    return (0);
}

```

```

error: 'void Sample::read()' is protected within this context

```

```

37 |     S.read();
    |           ^

```

```

main.cpp:22:6: note: declared protected here

```

```

22 | void Sample::read()
    |     ^~~~~~

```

Solution

Solution 1:-Calling Protected Member function from Public Member Function

```

void Sample::read() // Protected Member Function
{

```

```

    cout<<"I am protected class"<<endl;
}

void Sample::display() //Public Member Function
{
    read(); // Calling Protected Member Function from Public Member Function
    cout<<"I am a public class"<<endl;
}

```

Output of the following Program:-

I am protected class

I am a public class

Solution 2:- Using Friend Function

```

#include <iostream>

using namespace std;

class Member
{
    protected:

    int protected_mem=0; //protected Member

    public:

    friend int func(Member); //friend Function

};

int func (Member m)
{
    m.protected_mem=5; //friend Function accessing Protected Member
}

```

```

        return m.private_mem;
    }

    int main()
    {
        Member m;

        cout<<"Friend Function changed value of Protected Member::"<<func(m);

        return(0);
    }

```

Output:-

Friend Function changed value of Protected Member from 0 to :: 5

3) Private- Private

```

#include <iostream>

using namespace std;

class Sample
{
    private:
    void read();
    void display();
};

void Sample::read()
{
    cout<<"I am private class"<<endl;
}

```

```
}
```

```
void Sample::display()
```

```
{
```

```
    cout<<"I am a public class"<<endl;
```

```
}
```

```
int main()
```

```
{
```

```
    Sample S;
```

```
    S.display();//Generates Error as calling Private Member Function
```

```
    S.read();//Generated Error as Calling Private Member Function
```

```
    return (0);
```

```
}
```

```
error: 'void Sample::display()' is private within this context
```

```
34 |     S.display();
```

```
    |               ^
```

```
main.cpp:25:6: note: declared private here
```

```
25 | void Sample::display()
```

```
    |      ^~~~~~
```

```
main.cpp:35:12: error: 'void Sample::read\(\)' is private within this context
```

```
35 |     S.read();
```

```
    |               ^
```

```
main.cpp:20:6: note: declared private here
```

```
20 | void Sample::read()
```

```
    |      ^~~~~~
```


Solution:-

Program should contain 1 public function to access any Private Function

So, one of the Private Function should be converted into Public Member Function.

For Example:-

“FROM”

private:

void read();

void display();

“TO”

private:

void read();

public:

void display();

Outputs:-

I am private class

I am a public class