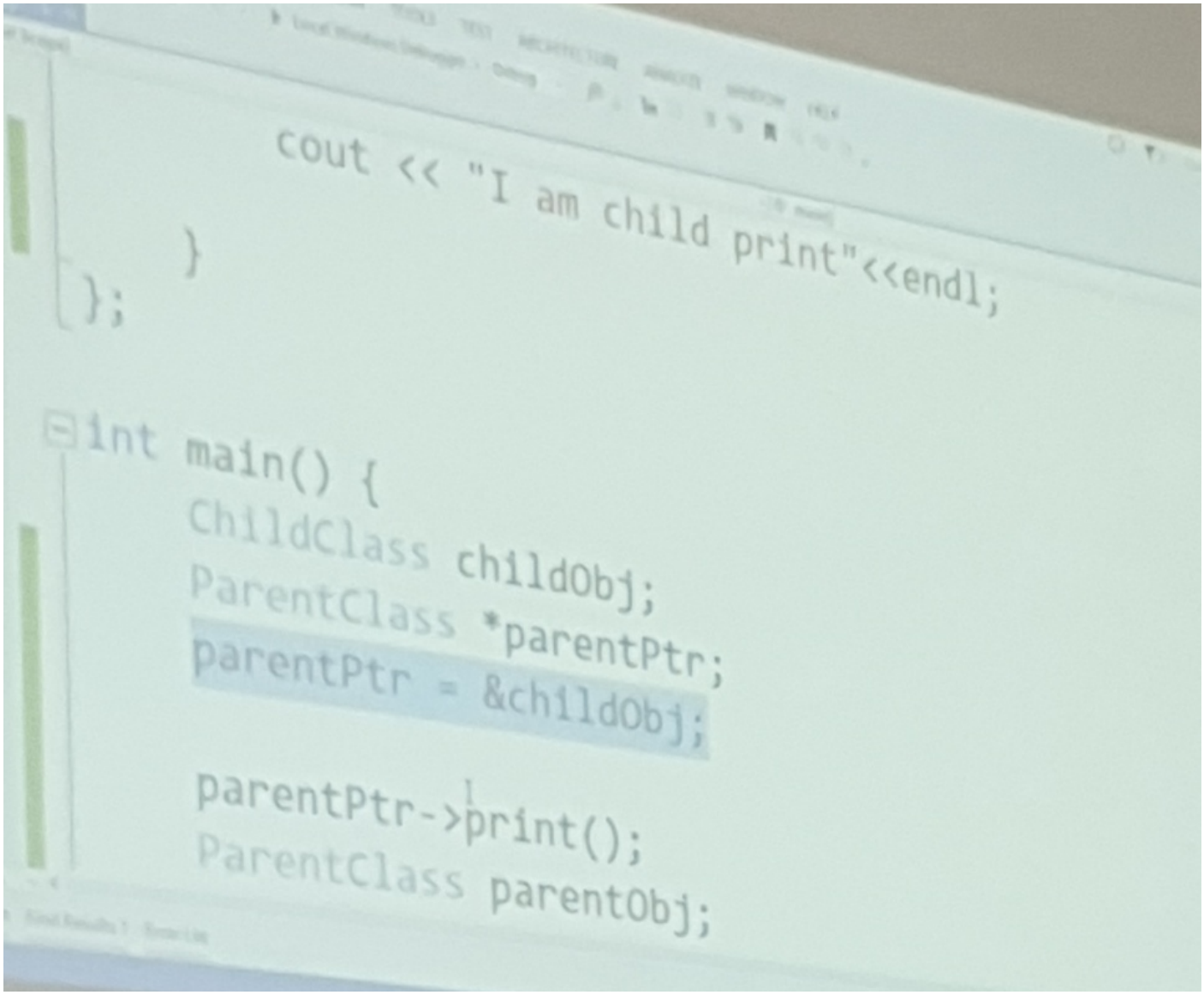
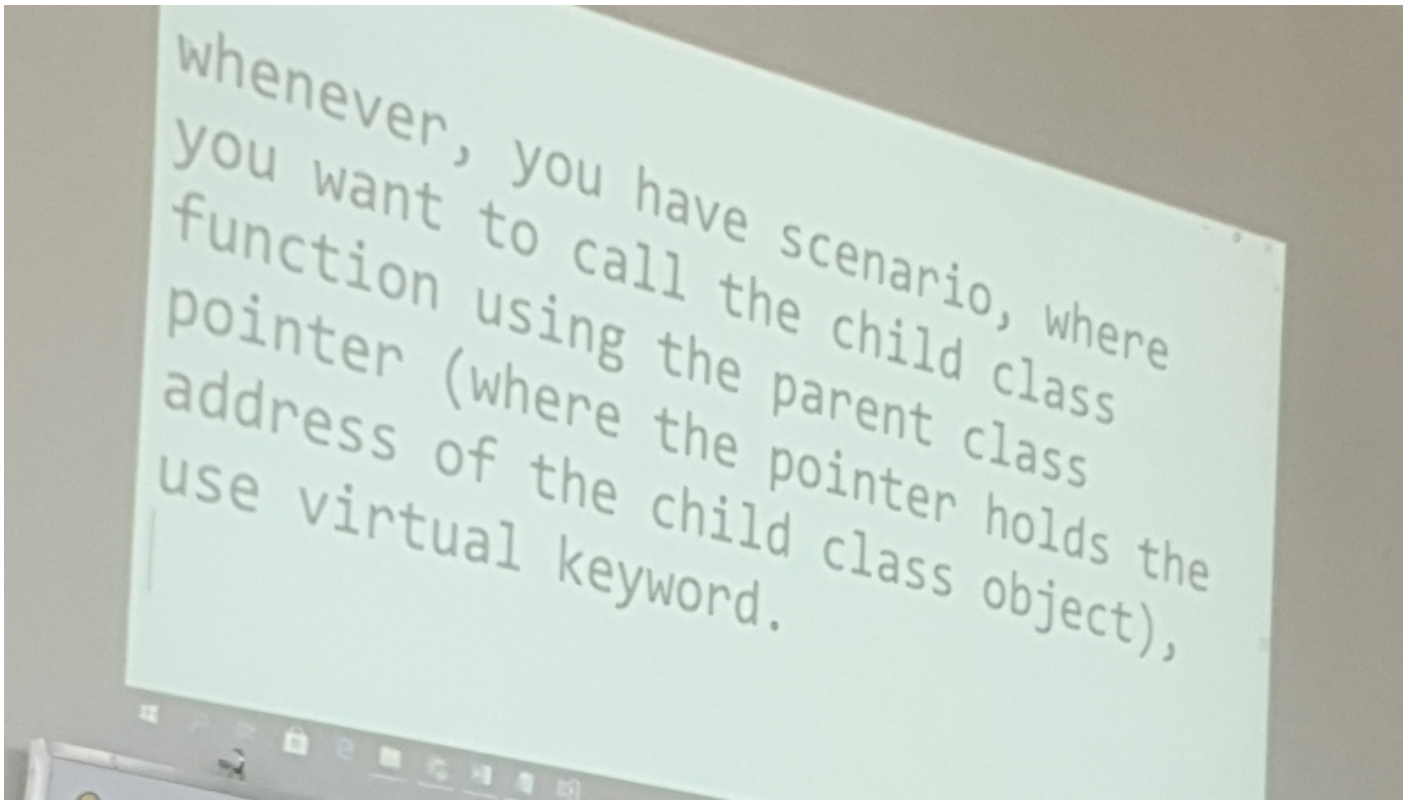
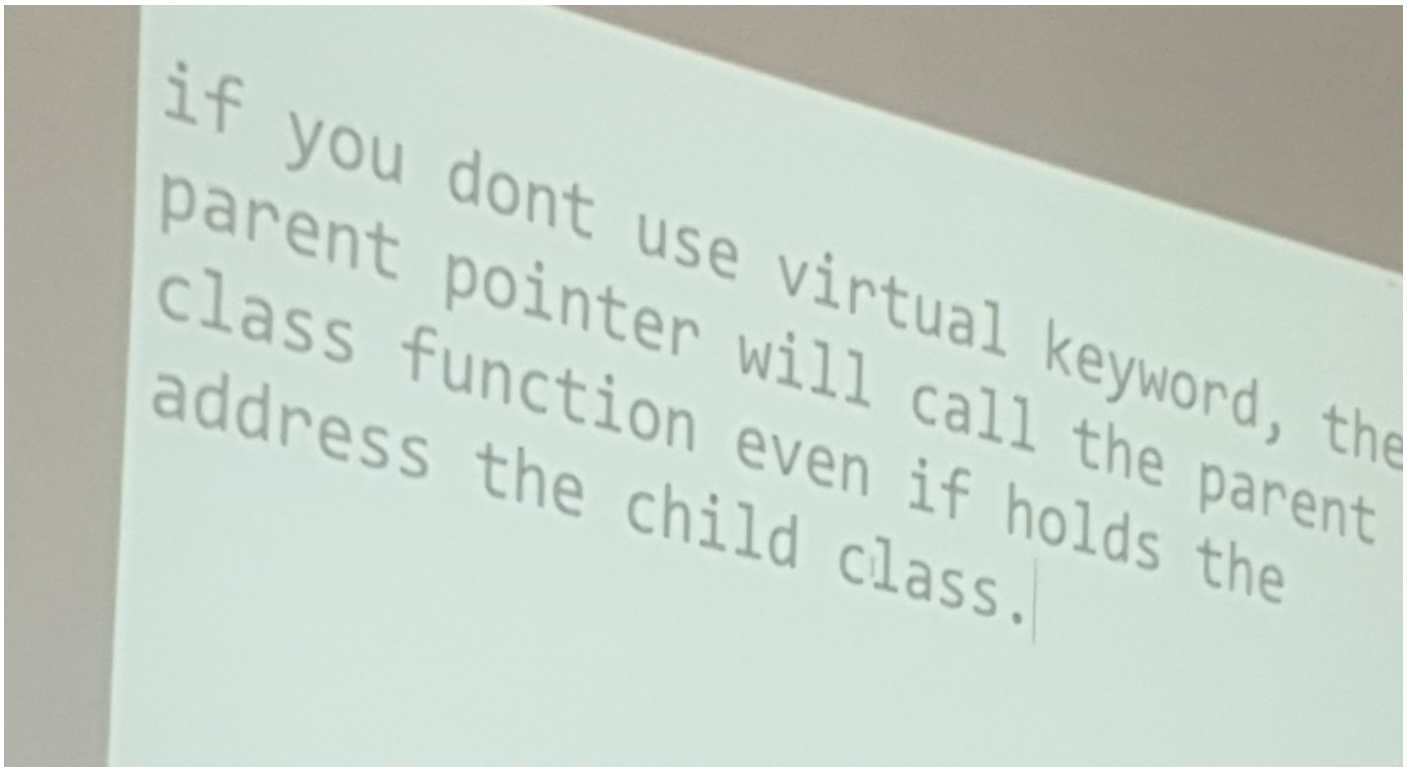
OOP(Function overriding + Virtual Function)

Function overriding

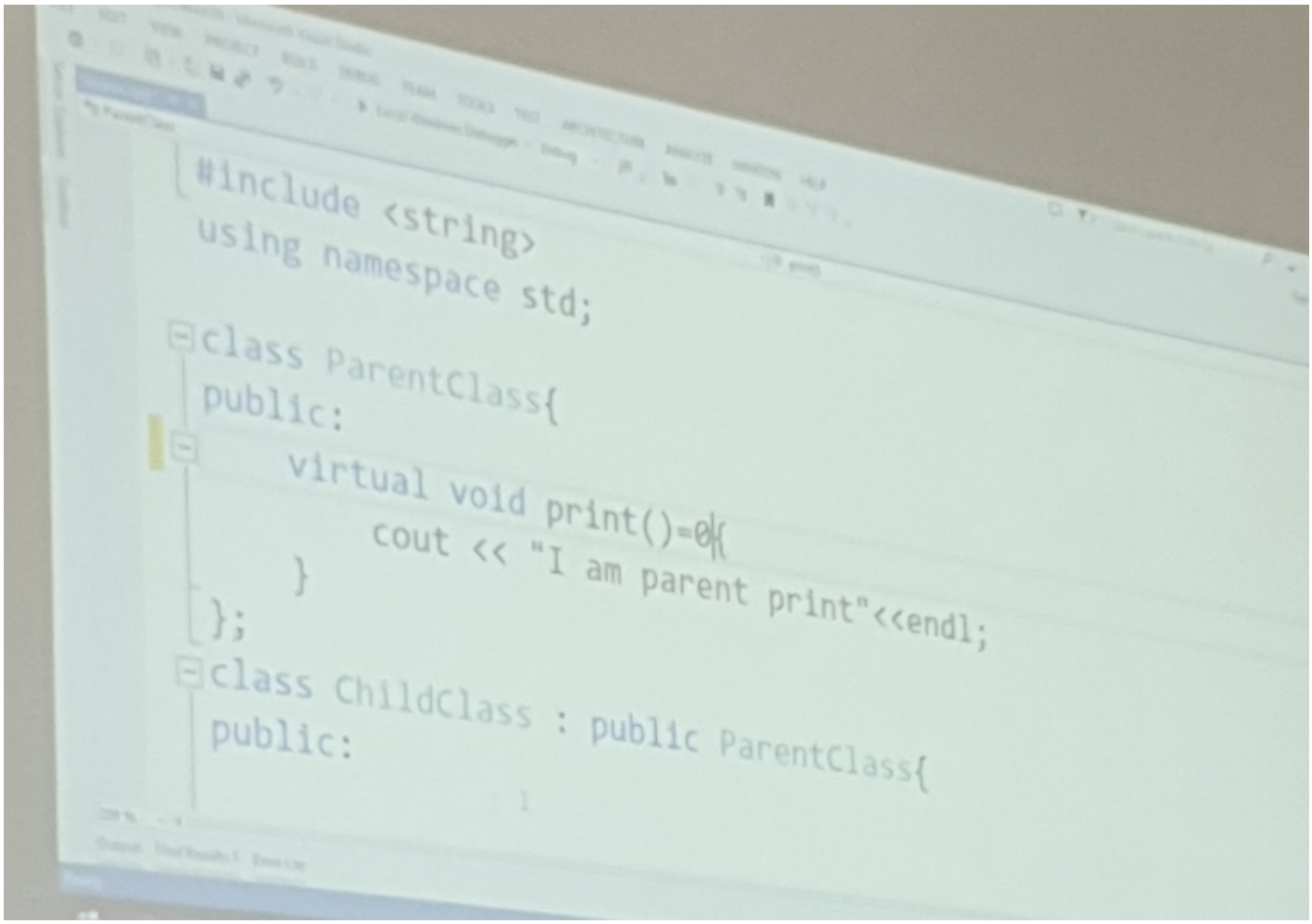


Virtual Function



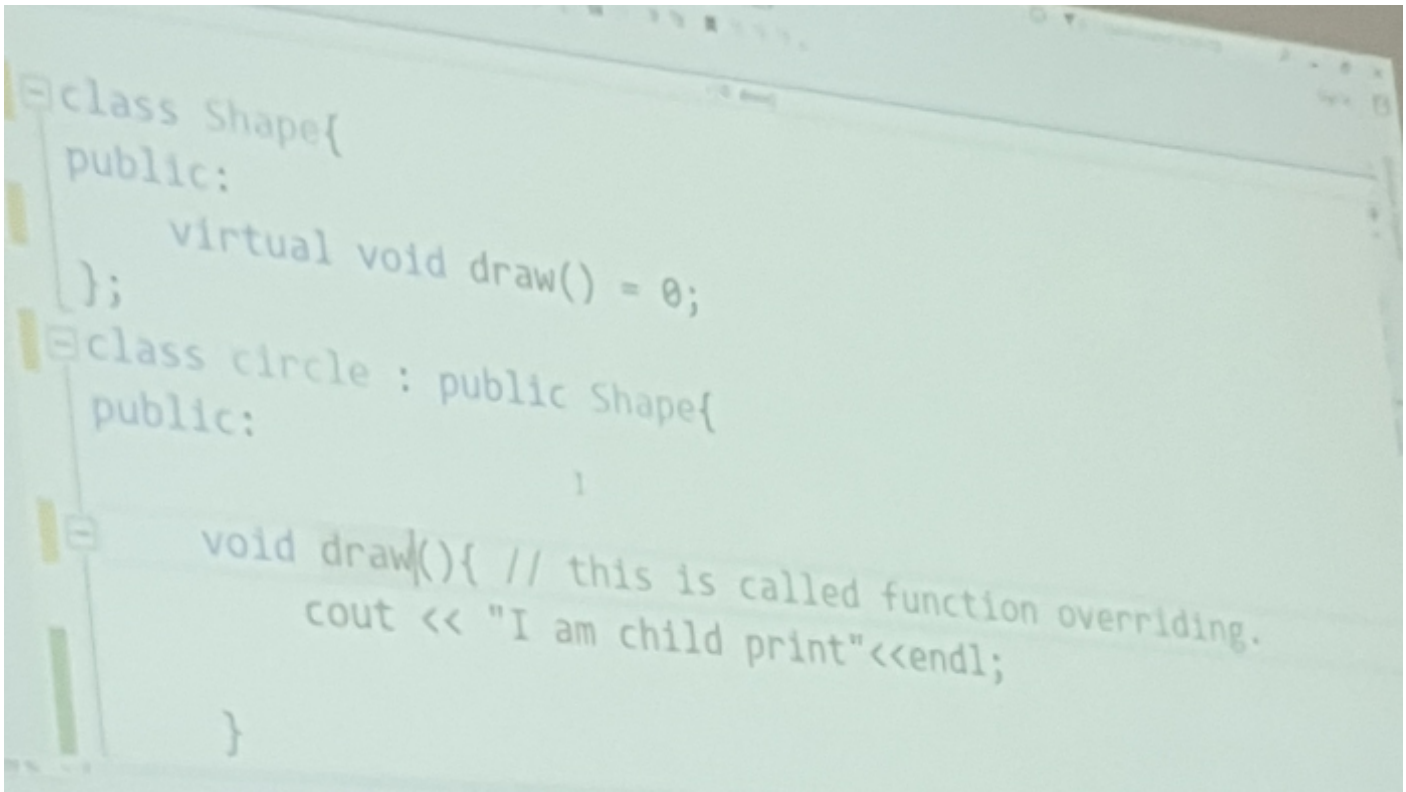


Pure Virtual Function....



```
#include <string>
using namespace std;

class ParentClass{
public:
    virtual void print()=0{
        cout << "I am parent print"<<endl;
    }
};
class ChildClass : public ParentClass{
public:
```



```
class Shape{
public:
    virtual void draw() = 0;
};
class circle : public Shape{
public:

    void draw(){ // this is called function overriding.
        cout << "I am child print"<<endl;

    }
```

Abstract class : The class frorm which we can not make any instance / object.

Concrete class : From which we can create objects.