

## 객체지향설계 #Week07

제출일 : 20.10.17

학번 이름 : 201902721 유찬희

GitHub 주소 : <https://github.com/HanCiHu/OOP>

homework1.

```
15 std::ostream& operator<<(std::ostream& os, AnyString& a)
16 {
17     os << a.getAnyString();
18     return os;
19 }
```

<< operator오버로딩 부분 코드

```
22 int main() {
23     std::ofstream out("testOveroding.txt");
24     AnyString a("Hello, this is operator overloading test!!!");
25     std::string output = a.getAnyString();
26     //out << output << std::endl;
27     out << a << std::endl;
28     return 0;
29 }
```

<< main 함수 부분

```
homework_07_01.cpp testOveroding.txt homework_07_02.cpp
week7 > testOveroding.txt
1 Stored String :: Hello, this is operator overloading test!!!
2
```

<< 실행결과

- <<연산자 뒤에 AnyString을 인식 할 수있도록 <<연산자를 오버로드 하였다.
- os 변수에 getAnyString 함수의 리턴값을 저장하고 os를 리턴해준다.

homework2.

```
1  #include <iostream>
2
3  bool funcA() {
4      int c;
5      std::cin >> c;
6      if (c < 10) //Let's assume this is exception
7          return false;
8      return true;
9  }
10 int main() {
11     try{
12         if (funcA()) {
13
14         }
15         else {
16             throw "Exception : Invalid Input!!";
17         }
18     }catch(const char* s){
19         std::cout << s << std::endl;
20     }
21     return 0;
22 }
23
```

<< homework2.cpp 코드

- catch의 매개변수에 string 변수를 넣어줄경우 오류가 발생해서 char\* 형으로 받아서 출력을 하게 만들었다.
- 사용자가 10보다 작은 수를 입력하게 되면 false가 리턴되고, false가 리턴되면 throw 줄의 코드가 실행되고, 그러면 catch안의 코드가 실행되게 된다.