

## - Example

An example of rvalue and lvalue references in modern C++.

### WE'LL COVER THE FOLLOWING ^

- Example
- Explanation

## Example #

```
// rvalueReference.cpp
#include <algorithm>
#include <iostream>
#include <string>

struct MyData{};

std::string function( const MyData & ) {
    return "lvalue reference";
}

std::string function( MyData && ) {
    return "rvalue reference";
}

int main(){

    std::cout << std::endl;

    MyData myData;

    std::cout << "function(myData): " << function(myData) << std::endl;
    std::cout << "function(MyData()): " << function(MyData()) << std::endl;
    std::cout << "function(std::move(myData)): " << function(std::move(myData)) << std::endl;

    std::cout << std::endl;
}
```



## Explanation #

# Explanation #

The code above is a simple example of rvalue and lvalue references.

- In line 22, `myData` is an lvalue since it has a name and address.
- In line 23, `MyData()` is an rvalue since it has neither a name nor a reference. Rather, this rvalue is a call to the default constructor of the `struct MyData`.
- In line 24, `std::move(myData)` creates an rvalue reference as well since you can neither determine the address of the `myData` nor the created object's name.

---

In the next lesson, we will learn about the differences between the copy semantic and the move semantic.