# - Example

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

## Rvalue/Lvalue references as arguments #

```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 #

The code above is a simple example of rvalue and lvalue references being used as arguments:

- In line 21, `myData` is an lvalue reference since it has a name and address.

- In line 22, `MyData()` is an rvalue reference since it does not have a name or an address. It is just a call to the default constructor of the struct `MyData`.

- In line 23, `std::move(myData)` creates an rvalue reference as well since we can neither determine the destination address of `myData`, nor the destination variable's name.

---

Next, we'll compare the semantics of **copy** and **move**.