

# Junior C++ developer

## Лекция 8

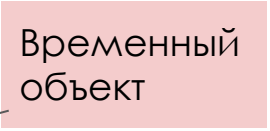
- move-семантика

# Временные объекты

- Создаются автоматически компилятором
- Не имеют имени
- Уничтожаются сразу же, как только в них исчезла необходимость

```
vector<string> strings;  
strings.push_back(string("hello"));
```

Временный  
объект

A pink rectangular box containing the text "Временный объект" (Temporary object) is positioned above the code. A black arrow points from the bottom-left corner of this box to the `string("hello")` argument in the `push_back` function call of the second line of code.

## Не временный объект

```
vector<string> strings;  
string tmp_str("hello");  
strings.push_back(tmp_str);
```

Компилятор копирует `tmp_str` и положить копию в вектор. Выполняется операция копирования.

## Временный объект

```
vector<string> strings;  
strings.push_back(string("hello"));
```

Для временного объекта не нужно делать копию, компилятор перемещает временный объект, не тратит при этом ресурсы на дорогостоящую операцию копирования.

## Не временный объект **l-value**

```
vector<string> strings;  
string tmp_str("hello");  
strings.push_back(tmp_str);
```

Компилятор копирует **tmp\_str** и положить копию в вектор.  
Выполняется операция копирования.

***l-value = r-value*** - ok

***r-value = l-value*** - ошибка

## Временный объект **r-value**

```
vector<string> strings;  
strings.push_back(string("hello"));
```

Для временного объекта не нужно делать копию, компилятор **перемещает** временный объект, не тратит при этом ресурсы на дорогостоящую операцию копирования.

# r-value

Для **r-value** объектов операции копирования заменяются операциями перемещения.

Перемещение намного быстрее и эффективнее копирования.

# Что если я хочу чтобы l-value вели себя как r-value?

Если я точно знаю что объект, который не является временным, на самом деле после операции копирования мне больше не нужен, я могу его переместить?

# move-семантика

Позволяет ***l-value*** объекту вести себя как ***r-value***.

# std::move

std::move(object)

- Принимает ссылку на **r-value** или **l-value** объект
- Возвращает **r-value** объект

```
std::vector<std::string> v;  
std::string str = "example";  
v.push_back(std::move(str)); /* str валидная переменная,  
но ее значение не определено */  
str.back(); /* неопределенное поведение (undefined behavior) если size() == 0:  
back() должен вызываться только перед !empty() */  
str.clear(); // OK, clear()
```



# Пример

```
#include <iostream>
#include <utility>
#include <vector>
#include <string>

int main()
{
    std::string str = "Hello";
    std::vector<std::string> v;

    // использование копирования
    v.push_back(str);
    std::cout << "After copy, str is " << str << "\n";

    // использование перемещения
    v.push_back(std::move(str));
    std::cout << "After move, str is " << str << "\n";

    std::cout << "The contents of the vector are " << v[0]
              << ", " << v[1] << "\n";
}
```