

# Видимость и инициализация переменных Конспект

# Области видимости переменных

Область видимости (англ. scope) — часть программы, где к объектам этой части можно обращаться по имени. Переменная видна от места объявления и до ближайшей закрывающей операторной скобки.

```
for (const string& word : query.plus_words) {  
    // тут пока не видно  
    if (word_to_document_freqs_.count(word) == 0) {  
        continue;  
    }  
    // и тут всё ещё нет  
    const double inverse_document_freq = ComputeWordInverseDocumentFreq(word);  
    // now you see me  
    for (const auto [document_id, term_freq] : word_to_document_freqs_.at(word)) {  
        const auto& document_data = documents_.at(document_id);  
        if (document_predicate(document_id, document_data.status, document_data.rating)) {  
            document_to_relevance[document_id] += term_freq * inverse_document_freq;  
        }  
    }  
}  
// now you don't
```

Переменная **word** объявлена в заголовке цикла, поэтому видна во всём цикле, но не снаружи. Область видимости параметров функции — это всё тело функции. Объявление переменные без учёта области видимости приводит к ошибкам. Чтобы код работал, был понятным и легко тестировался, сразу объявляйте переменную максимально глубоко.

## Инициализация переменных

Инициализируйте все переменные и поля числовых типов:

- целочисленные — int, size\_t, int64\_t;
- вещественные — float, double;



- char;
- bool;
- enum.

Создание локальных переменных — резервирование области памяти нужного размера. С точки зрения времени выполнения программы это бесплатно.

Создавая без дополнительной информации более сложные объекты, такие как строки, векторы, словари, вы получите пустой объект.

Поля самостоятельно созданных структур и классов по умолчанию инициализируются подобно локальным переменным. В стандартных классах явная инициализация необязательна.



```
cout << "Каждый может стать" << endl;
```