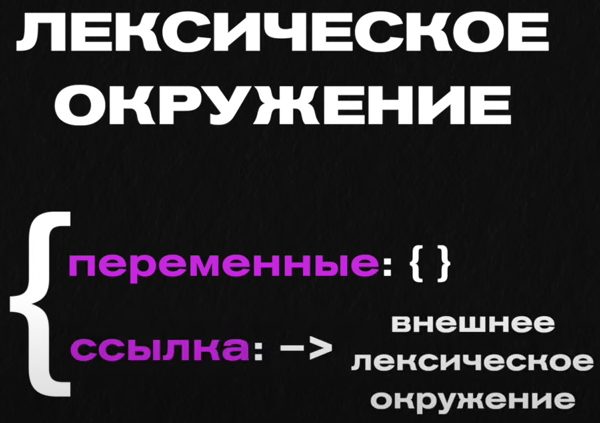
**Замыкания**

**Замыкание** – это способность функции в js запоминать лексическое окружение, в котором она была создана. Т.е. хранить в себе ссылку на это окружение.

Лексическое окружение – это невидимый (скрытый) объект, который есть у любого блока, скрипта или функции в js.

Данный объект состоит из двух частей:

1. Объект с переменными в текущей области видимости
2. Ссылка на внешнее лексическое окружение



Объяснение

*const* x = 1;

*const* y = *function*() {

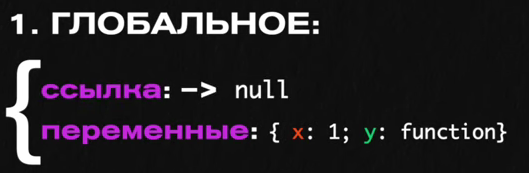
*const* i = 'Hi';

    console.log(i);

}

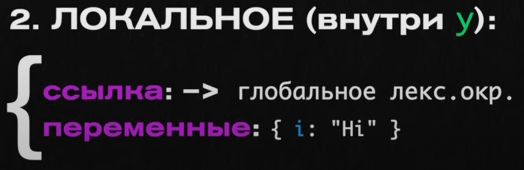
В данном коде есть 2 лексических окружения:

1. Глобальное (над глобальным окружением ничего нет поэтому ссылка null)



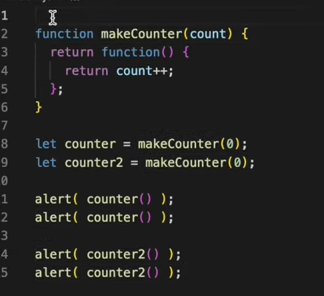
Имеет доступ только к своим переменным

1. Локальное (это окружение будет создано только в момент вызова функции)



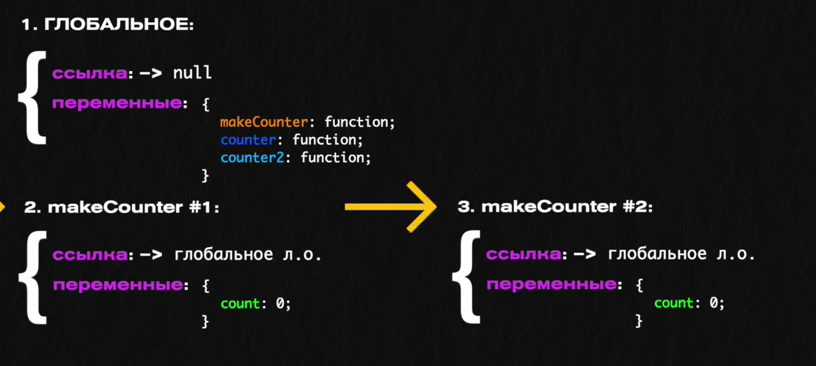
Имеет доступ ко всем переменным

Пример 1



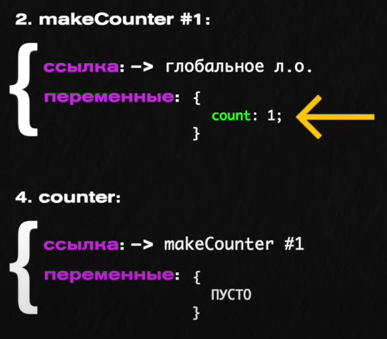
* На первом этапе создается глобальное лексическое окружение
* На 8 строке вызов функции makeCounter, создается новое лексическое окружение. Каждый аргумент функции это переменная count: 0
* На 10 строке функции makeCounter, создается новое лексическое окружение. Каждый аргумент функции это переменная count: 0

Лексические окружение на 8 строке и 10 это разные объекты



* На 12 строке вызываем функцию alert и вызывается функция counter. В counter содержится функция, которая увеличивает переменную count++.

Создается лексическое окружение, которое имеет ссылку на лексическое окружение makeCounter. Переменных у этого окружения нет. Для получения данной переменной мы обращаемся к окружению makeCounter у которой есть переменная count.



Пример 2