



Занятие 4

Тема: Функции

Цель:

Познакомиться:

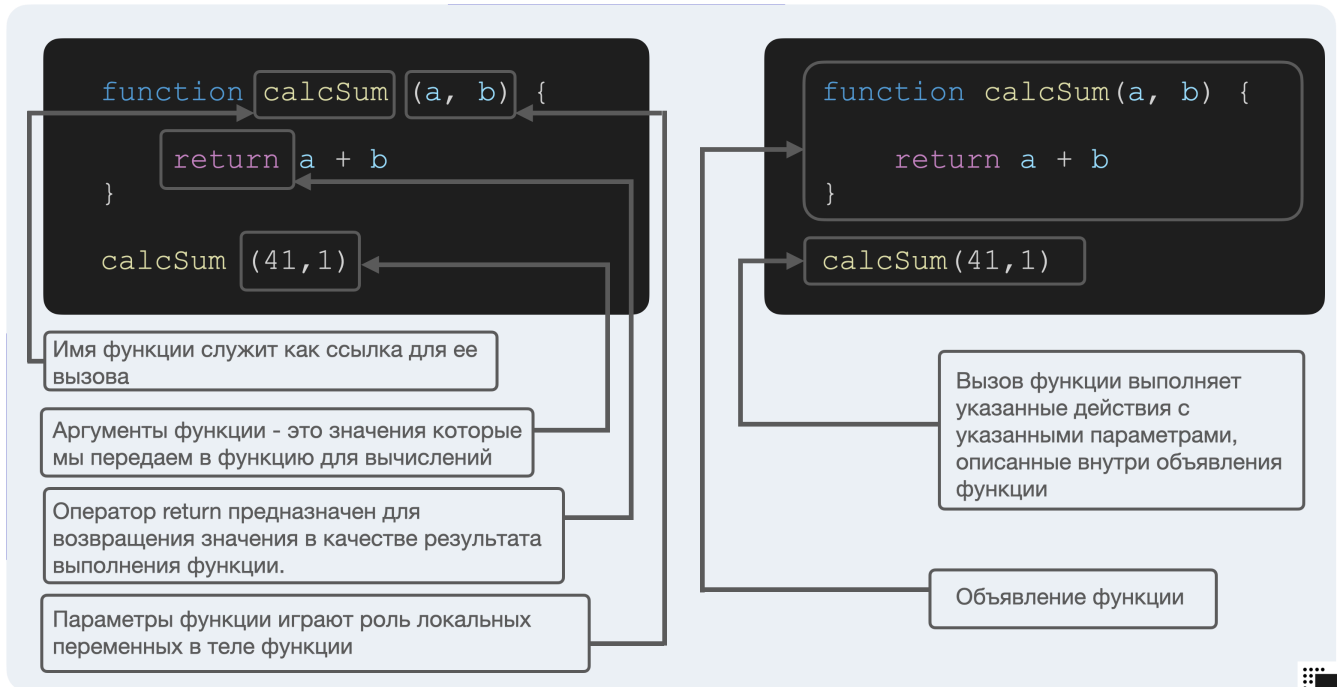
- Философия и применение функций
- Синтаксис
- Локальные и внешние переменные
- Функциональные выражения

План занятия:

- Философия функций - зачем нужны и где использовать
- Функции:
 - Объявление
 - Вызов
 - Имя
 - Тело
 - Аргументы
 - Параметры
 - Ключевое слово return
- Локальные и внешние переменные
- Функциональное выражение - синтаксис, особенности, способы применения.

Конспект:

Функции:



Рекомендации по именованию функций:

Recomendation for naming functions

Step 1

Follow this rules 🧑💻

- Use **camelCase**
- Function name must start with a **verb**
- Do not use **transliteration**
- Do not use **abbreviations**
- Use **SID** rules:
 - **Short.** A name must not take long to type and, therefore, remember;

Step 2

Follow **A/HC/LC** Pattern

Name	Prefix	Action (A)	High context (HC)	Low context (LC)
getUser		get	User	
getUserMessages		get	User	Messages
handleClickOutside		handle	Click	Outside
shouldDisplayMessage	should	Display	Message	

Step 3

You stuck ? Try use this name examples

Prefixes

is has can should _ # on

Action verbs

display print render remove delete reset update sort validate compose init get set calc
fetch handle bind apply post load

Внешние и локальные переменные:

- Функция может работать как с внешними так и с локальными переменными

Функциональные выражения:

- Функциональные выражения в JavaScript не поднимаются (hoisting), в отличие от объявленных функций. Вы не можете использовать функциональные выражения прежде, чем вы их определили.
- Главным отличием между Function Declaration является *имя функции*, которое в случае функциональных выражений может быть опущено для создания *анонимных* функций.

Дополнительные материалы:

- Домашнее задание в файле **hw-4.md**



< Teach
Me
Skills / >

