ВВЕДЕНИЕ

В настоящее время в связи с развитием технологий так же развиваются и языки программирования, изменяются, обновляются, модернизируются и улучшаются разработчиками.

Но всё же есть такие языки, которые являются самыми популярными и используемые для каких-либо целей, от вёрстки простейших сайтов, до разработки разнообразных, сложных по структуре, приложений и игр.

Одним из самых популярных языков для разработки веб-приложений, является язык программирования PHP (Hypertext Processor – процессор гипертекста). Данный язык является скриптовым (сценарным), то есть он состоит из кратких описаний действий выполняемых системой. А сценарий в свою очередь – это программа, имеющая дело с готовыми программными компонентами.

В настоящее время PHP поддерживается подавляющим большинством хостинг провайдеров и является одним из лидеров среди языков, применяющихся для создания динамических веб-сайтов.

Так же развивается не менее популярный язык JavaScript -[мультипарадигменный](https://ru.wikipedia.org/wiki/%D0%9C%D1%83%D0%BB%D1%8C%D1%82%D0%B8%D0%BF%D0%B0%D1%80%D0%B0%D0%B4%D0%B8%D0%B3%D0%BC%D0%B5%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) [язык программирования](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) ([программирование](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) с одновременным использованием множества [парадигм](https://ru.wikipedia.org/wiki/%D0%9F%D0%B0%D1%80%D0%B0%D0%B4%D0%B8%D0%B3%D0%BC%D0%B0_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)).

JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам.

Данный язык обладает такими архитектурными чертами как: динамическая типизация, слабая типизация, автоматическое управление памятью, прототипное программирование, а так же функции (в качестве объектов первого класса).

На JavaScript оказали влияние многие языки, при разработке была цель сделать язык похожим на Java, но при этом лёгким для использования непрограммистами. Языком JavaScript не владеет какая-либо компания или организация, что отличает его от ряда языков программирования, используемых в веб-разработке.

Не смотря на то, что PHP и JavaScript по своей структуре и синтаксису являются абсолютно различными, их можно объединять. К примеру, написанный PHP код с лёгкостью дополняется кодом JavaScript, что является, может и не самым удобным, но всё же максимально эффективным в функциональной части программы.

Эти два скриптовых языка свободно между собой взаимодействуют что облегчает работу даже в таких элементарных вещах как передача переменных и тому подобному.

**PHP базы данных и синтаксис**

Данный язык обладает большим количеством возможностей, которые определяется наличием большого набора встроенных средств для разработки веб-приложений.

Основные из них:

* автоматическое извлечение POST и GET -параметров, а также переменных окружения веб-сервера в предопределённые массивы;
* взаимодействие с большим количеством различных систем управления базами данных MySQL, MySQLi, SQLite, PostgreSQL, Oracle(OCI8) многие другие;
* автоматизированная отправка [HTTP-заголовков](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%B3%D0%BE%D0%BB%D0%BE%D0%B2%D0%BA%D0%B8_HTTP);
* работа с HTTP-авторизацией;
* работа с [cookies](https://ru.wikipedia.org/wiki/Cookies) и сессиями;
* работа с локальными и удалёнными файлами, [сокетами](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D0%BA%D0%B5%D1%82_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D1%8B%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81));
* обработка файлов, загружаемых на сервер;
* работа с [XForms](https://ru.wikipedia.org/wiki/XForms).

В настоящее время PHP используется сотнями тысяч разработчиков. Согласно рейтингу корпорации TIOBE, базирующемся на данных поисковых систем, в мае 2016 года PHP находился на 6-м месте среди языков программирования. К крупнейшим сайтам, использующим PHP, относятся [Facebook](https://ru.wikipedia.org/wiki/Facebook), [Wikipedia](https://ru.wikipedia.org/wiki/Wikipedia) и др.

Входит в [LAMP](https://ru.wikipedia.org/wiki/LAMP) — распространённый набор программного обеспечения для создания и [хостинга](https://ru.wikipedia.org/wiki/%D0%A5%D0%BE%D1%81%D1%82%D0%B8%D0%BD%D0%B3) [веб-сайтов](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%B0%D0%B9%D1%82) ([Linux](https://ru.wikipedia.org/wiki/Linux), [Apache](https://ru.wikipedia.org/wiki/Apache_HTTP_Server), [MySQL](https://ru.wikipedia.org/wiki/MySQL), PHP).

### Типы данных

PHP является языком программирования с [динамической типизацией](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%BD%D0%B0%D0%BC%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D1%82%D0%B8%D0%BF%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F), не требующим указания типа при объявлении переменных, равно как и самого объявления переменных. Преобразования между скалярными типами зачастую осуществляются неявно без дополнительных усилий.

К скалярным типам данных относятся:

* [целочисленный тип](https://ru.wikipedia.org/wiki/%D0%A6%D0%B5%D0%BB%D1%8B%D0%B9_%D1%82%D0%B8%D0%BF) (integer)
* [число с плавающей точкой](https://ru.wikipedia.org/wiki/%D0%A7%D0%B8%D1%81%D0%BB%D0%B0_%D1%81_%D0%BF%D0%BB%D0%B0%D0%B2%D0%B0%D1%8E%D1%89%D0%B5%D0%B9_%D0%B7%D0%B0%D0%BF%D1%8F%D1%82%D0%BE%D0%B9) (float, double)
* [логический тип](https://ru.wikipedia.org/wiki/%D0%9B%D0%BE%D0%B3%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D1%82%D0%B8%D0%BF) (boolean)
* [строковый тип](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D1%80%D0%BE%D0%BA%D0%BE%D0%B2%D1%8B%D0%B9_%D1%82%D0%B8%D0%BF) (string)

К нескалярным типам относятся:

* [массив](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D0%B4%D0%B5%D0%BA%D1%81%D0%BD%D1%8B%D0%B9_%D0%BC%D0%B0%D1%81%D1%81%D0%B8%D0%B2) (array)
* [объект](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) (object)
* внешний ресурс (resource)
* неопределенное значение (null)

К псевдотипамотносятся:

* [mixed](https://ru.wikipedia.org/w/index.php?title=Mixed&action=edit&redlink=1) любой тип
* [number](https://ru.wikipedia.org/w/index.php?title=Number&action=edit&redlink=1) число (integer либо float)
* [callback](https://ru.wikipedia.org/wiki/Callback_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) (string или [анонимная функция](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%BC%D1%8B%D0%BA%D0%B0%D0%BD%D0%B8%D0%B5_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)))
* [void](https://ru.wikipedia.org/wiki/Void) отсутствие параметров

Диапазон целых чисел (integer) в PHP зависит от [платформы](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%BF%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B0) (обычно, это диапазон 32-битных знаковых целых чисел, то есть, от −2147483648 до 2147483647). Числа можно задавать в десятичной, [восьмеричной](https://ru.wikipedia.org/wiki/%D0%92%D0%BE%D1%81%D1%8C%D0%BC%D0%B5%D1%80%D0%B8%D1%87%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%81%D1%87%D0%B8%D1%81%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F) и [шестнадцатеричной](https://ru.wikipedia.org/wiki/%D0%A8%D0%B5%D1%81%D1%82%D0%BD%D0%B0%D0%B4%D1%86%D0%B0%D1%82%D0%B5%D1%80%D0%B8%D1%87%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%81%D1%87%D0%B8%D1%81%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F) системах счисления. Диапазон вещественных чисел (double) также зависит от платформы (для 32-битной архитектуры диапазон позволяет оперировать числами от ±1.7×10−308 до ±1.7×10+308).

PHP предоставляет разработчикам [логический тип](https://ru.wikipedia.org/wiki/%D0%9B%D0%BE%D0%B3%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D1%82%D0%B8%D0%BF) (boolean), способный принимать только два значения TRUE («истина») и FALSE («ложь»). При преобразовании в логический тип число 0, пустая строка, ноль в строке «0», NULL и пустой массив считаются равными FALSE. Все остальные значения автоматически преобразуются в TRUE.

Специальный тип null предназначен для переменных без определённого значения. Единственным значением данного типа является константа NULL. Тип null принимают неинициализированные переменные, переменные инициализированные константой NULL, а также переменные, удалённые при помощи конструкции unsert().

Ссылки на внешние ресурсы имеют тип «ресурс» (resource). Переменные данного типа, как правило, представляют собой [дескриптор](https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D1%81%D0%BA%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D1%80_(%D1%8F%D0%B7%D1%8B%D0%BA%D0%B8_%D1%80%D0%B0%D0%B7%D0%BC%D0%B5%D1%82%D0%BA%D0%B8)), позволяющий управлять внешними объектами, такими как файлы, динамические изображения, результирующие таблицы базы данных и т. п.

[Массивы](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D0%B4%D0%B5%D0%BA%D1%81%D0%BD%D1%8B%D0%B9_%D0%BC%D0%B0%D1%81%D1%81%D0%B8%D0%B2) (array) поддерживают числовые и строковые ключи и являются [гетерогенными](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D0%B4%D0%B5%D0%BA%D1%81%D0%BD%D1%8B%D0%B9_%D0%BC%D0%B0%D1%81%D1%81%D0%B8%D0%B2#%D0%A1%D0%BF%D0%B5%D1%86%D0%B8%D1%84%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B5_%D1%82%D0%B8%D0%BF%D1%8B_%D0%BC%D0%B0%D1%81%D1%81%D0%B8%D0%B2%D0%BE%D0%B2). Массивы могут содержать значения любых типов, включая другие массивы. Порядок элементов и их ключей сохраняется. Не совсем корректно называть php-массивы массивами, на самом деле это, скорее всего, упорядоченный [хеш](https://ru.wikipedia.org/wiki/%D0%90%D1%81%D1%81%D0%BE%D1%86%D0%B8%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D1%8B%D0%B9_%D0%BC%D0%B0%D1%81%D1%81%D0%B8%D0%B2#PHP). Возможно неожиданное поведение при использовании цикла for со счетчиком вместо foreach. Так, например, при сортировке массива с численными индексами функциями из стандартной библиотеки, сортируются и ключи тоже.

PHP обладает полной поддержкой [ООП](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5). Работа с классами была оптимизирована и теперь такой код работает достаточно быстро.

[Класс](https://ru.wikipedia.org/wiki/%D0%9A%D0%BB%D0%B0%D1%81%D1%81_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) в PHP объявляется с помощью ключевого слова class Методы и свойства класса могут быть общедоступными (public, по умолчанию), защищёнными (protected) и скрытыми (private). PHP поддерживает все три основных механизма ООП - [инкапсуляцию](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D0%BA%D0%B0%D0%BF%D1%81%D1%83%D0%BB%D1%8F%D1%86%D0%B8%D1%8F_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), [полиморфизм подтипов](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%BB%D0%B8%D0%BC%D0%BE%D1%80%D1%84%D0%B8%D0%B7%D0%BC_%D0%BF%D0%BE%D0%B4%D1%82%D0%B8%D0%BF%D0%BE%D0%B2) и [наследование](https://ru.wikipedia.org/wiki/%D0%9D%D0%B0%D1%81%D0%BB%D0%B5%D0%B4%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) (родительский класс указывается с помощью ключевого слова extends после имени класса). Поддерживаются [интерфейсы](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_(%D0%9E%D0%9E%D0%9F)) (ставятся в соответствие с помощью implements). Разрешается объявление [финальных](https://ru.wikipedia.org/wiki/%D0%9B%D0%B8%D1%81%D1%82%D0%BE%D0%B2%D0%BE%D0%B9_%D0%BA%D0%BB%D0%B0%D1%81%D1%81), [абстрактных методов](https://ru.wikipedia.org/wiki/%D0%90%D0%B1%D1%81%D1%82%D1%80%D0%B0%D0%BA%D1%82%D0%BD%D1%8B%D0%B9_%D0%BC%D0%B5%D1%82%D0%BE%D0%B4) и классов. [Множественное наследование](https://ru.wikipedia.org/wiki/%D0%9D%D0%B0%D1%81%D0%BB%D0%B5%D0%B4%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)#%D0%9C%D0%BD%D0%BE%D0%B6%D0%B5%D1%81%D1%82%D0%B2%D0%B5%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BD%D0%B0%D1%81%D0%BB%D0%B5%D0%B4%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) классов не поддерживается, однако класс может реализовывать несколько интерфейсов. Для обращения к методам родительского класса используется ключевое слово parent.

**Скрипт**

PHP по своему синтаксису, схож с языком Cи. А так же имеются такие элементы как, ассоциативные массивы и цикл foreach, которые были позаимствованы из PERL.

Для работы программы не требуется описывать каких-либо переменных, используемых модулей и т. п. Любая программа может начинаться непосредственно с оператора PHP.

Простейшая программа на PHP выглядит следующим образом:

<?php

Echo “Hello, world!”;

<?

Но даже и это можно написать более коротко:

<?= “Hello, world!” ?>

Исходя из того что это скриптовый язык, его можно и даже нужно использовать совместно с HTML если это уместно, то есть при вёрстке сайтов. При этом для того чтобы использовать их совместно, необходимо файл с кодом данного сайта создать формата PHP, а не HTML.

PHP-скрипты обычно обрабатываются [интерпретатором](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D0%BF%D1%80%D0%B5%D1%82%D0%B0%D1%82%D0%BE%D1%80) в порядке, обеспечивающем [мультиплатформенность](https://ru.wikipedia.org/wiki/%D0%9A%D1%80%D0%BE%D1%81%D1%81%D0%BF%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B5%D0%BD%D0%BD%D0%BE%D1%81%D1%82%D1%8C) разработанного приложения:

1. [лексический анализ](https://ru.wikipedia.org/wiki/%D0%9B%D0%B5%D0%BA%D1%81%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D0%B0%D0%BD%D0%B0%D0%BB%D0%B8%D0%B7) [исходного кода](https://ru.wikipedia.org/wiki/%D0%98%D1%81%D1%85%D0%BE%D0%B4%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%B4) и генерация [лексем](https://ru.wikipedia.org/wiki/%D0%9B%D0%B5%D0%BA%D1%81%D0%B5%D0%BC%D0%B0_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)),
2. [синтаксический анализ](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D0%B0%D0%BD%D0%B0%D0%BB%D0%B8%D0%B7) полученных лексем,
3. генерация [байт-кода](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B9%D1%82-%D0%BA%D0%BE%D0%B4),
4. выполнение байт-кода интерпретатором (без создания [исполняемого файла](https://ru.wikipedia.org/wiki/%D0%98%D1%81%D0%BF%D0%BE%D0%BB%D0%BD%D1%8F%D0%B5%D0%BC%D1%8B%D0%B9_%D1%84%D0%B0%D0%B9%D0%BB)).

Для увеличения быстродействия приложений возможно использование специального [программного обеспечения](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5), так называемых [акселераторов](https://ru.wikipedia.org/wiki/%D0%90%D0%BA%D1%81%D0%B5%D0%BB%D0%B5%D1%80%D0%B0%D1%82%D0%BE%D1%80_PHP). Принцип их работы заключается в [кэшировании](https://ru.wikipedia.org/wiki/%D0%9A%D1%8D%D1%88%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) однажды сгенерированного байт-кода в [памяти](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%B0%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C) и/или на [диске](https://ru.wikipedia.org/wiki/%D0%96%D1%91%D1%81%D1%82%D0%BA%D0%B8%D0%B9_%D0%B4%D0%B8%D1%81%D0%BA), таким образом, из процесса работы приложения исключаются этапы 1-3, что в общем случае ведёт к значительному ускорению работы.

Важной особенностью является то, что разработчику нет необходимости заботиться о распределении и [освобождении](https://ru.wikipedia.org/wiki/%D0%A1%D0%B1%D0%BE%D1%80%D0%BA%D0%B0_%D0%BC%D1%83%D1%81%D0%BE%D1%80%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) памяти. Ядро PHP реализует средства для автоматического управления памятью; вся выделенная память возвращается системе после завершения работы скрипта.

Интерпретатор состоит из [ядра](https://ru.wikipedia.org/wiki/Zend_Engine) и подключаемых [модулей](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D1%83%D0%BB%D1%8C_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), *«расширений»*, представляющих собой динамические [библиотеки](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)). Расширения позволяют дополнить базовые возможности языка, предоставляя возможности для работы с [базами данных](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), [сокетами](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D0%BA%D0%B5%D1%82_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D1%8B%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81)), динамической [графикой](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%B3%D1%80%D0%B0%D1%84%D0%B8%D0%BA%D0%B0), [криптографическими](https://ru.wikipedia.org/wiki/%D0%9A%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B3%D1%80%D0%B0%D1%84%D0%B8%D1%8F) библиотеками, документами формата [PDF](https://ru.wikipedia.org/wiki/PDF) и тому подобным. Любой желающий может разработать своё собственное расширение и подключить его. Существует огромное количество расширений, как стандартных, так и созданных сторонними компаниями и энтузиастами, однако в стандартную поставку входит лишь несколько десятков хорошо зарекомендовавших себя. Множество расширений доступно в репозитории [PECL](https://ru.wikipedia.org/wiki/PECL).

Интерпретатор PHP имеет специальный [конфигурационный файл](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D1%84%D0%B8%D0%B3%D1%83%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D1%8B%D0%B9_%D1%84%D0%B0%D0%B9%D0%BB) –php\_ini, содержащий множество настроек, изменение которых влияет на поведение интерпретатора. Имеется возможность отключить использование ряда функций, изменить ограничения на используемую скриптом оперативную память, время выполнения, объём загружаемых файлов, настроить журналирование ошибок, работу с сессиями и почтовыми сервисами, подключить дополнительные расширения, а также многое другое. Возможное дробление большого конфигурационного файла на части. Например, широко распространена практика вынесения настроек расширений в отдельные файлы. Параметры интерпретатора могут быть переопределены в файлах конфигурации [HTTP-сервера](https://ru.wikipedia.org/wiki/HTTP-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80) (например, htaccess в [Apache](https://ru.wikipedia.org/wiki/Apache_HTTP_Server)) или в самом скрипте во время выполнения при помощи команды ini\_set.

SAPI — это внешний [уровень абстракции](https://ru.wikipedia.org/wiki/%D0%A3%D1%80%D0%BE%D0%B2%D0%B5%D0%BD%D1%8C_%D0%B0%D0%B1%D1%81%D1%82%D1%80%D0%B0%D0%BA%D1%86%D0%B8%D0%B8_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), предназначенный, для встраивания интерпретатора в другие приложения и отвечает за его работу (запуск, остановка, передача скриптов на исполнение, доступ к внешним данным). Существует несколько основных SAPI, определяющих способы запуска и использования PHP:

1. В качестве модуля к [веб-серверу](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80) (например, для [Apache](https://ru.wikipedia.org/wiki/Apache_HTTP_Server) модуль mod\_php). В этом случае интерпретатор PHP выполняется в [окружении](https://ru.wikipedia.org/w/index.php?title=%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BE%D0%BA%D1%80%D1%83%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5&action=edit&redlink=1) [процесса](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) веб-сервера. Веб-сервер управляет количеством запущенных процессов PHP и сообщает им, какие скрипты требуется исполнить.
2. [CGI](https://ru.wikipedia.org/wiki/CGI) SAPI. Использование CGI подразумевает запуск нового процесса для обработки каждого запроса. Для исполнения PHP скрипта веб-сервер запускает ./php-cgi /path/to/script.php. Сам принцип такого использования подразумевает, что интерпретатор PHP исполняет только один скрипт, после чего заканчивает свою работу. Затраты на запуск процесса интерпретатора и его инициализацию очень часто сопоставимы или даже превышают затраты на исполнение PHP ю-скрипта. Для решения этой проблемы в CGI SAPI был введён режим FastCGI. В этом режиме PHP интерпретатор запускается как независимый сервер, обрабатывающий входящие запросы на исполнение PHP скриптов по протоколу FastCGI, что позволяет ему работать с любым веб-сервером, поддерживающим этот протокол.
3. FPM SAPI, известный как php-fpm — это другая реализация протокола FastCGI. Создан изначально Андреем Нигматулиным как отдельный патч для использования в социальной сети Badoo. Данная реализация решала ряд проблем, которые мешали использованию CGI/FastCGI SAPI. В частности, появилась возможность перезапуска пула интерпретаторов PHP без потери запросов, запуск нескольких пулов под разными пользователями, аварийный перезапуск интерпретаторов в случае проблем с ними и ещё несколько полезных дополнений. В дальнейшем над патчем работали несколько человек, был добавлен режим динамического управления числом запущенных процессов PHP (по принципу управления числом процессов в веб-сервере Apache), и, начиная с версии PHP 5.3.3, php-fpm был включён в PHP как отдельное SAPI.
4. В качестве [скрипта командной строки](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BA%D0%BE%D0%BC%D0%B0%D0%BD%D0%B4%D0%BD%D0%BE%D0%B9_%D1%81%D1%82%D1%80%D0%BE%D0%BA%D0%B8) (CLI SAPI), являющегося [исполняемым файлом](https://ru.wikipedia.org/wiki/%D0%98%D1%81%D0%BF%D0%BE%D0%BB%D0%BD%D1%8F%D0%B5%D0%BC%D1%8B%D0%B9_%D1%84%D0%B0%D0%B9%D0%BB), который вызывается пользователем из [командной строки](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BA%D0%BE%D0%BC%D0%B0%D0%BD%D0%B4%D0%BD%D0%BE%D0%B9_%D1%81%D1%82%D1%80%D0%BE%D0%BA%D0%B8); скрипт выполняется в окружении вызвавшего пользователя. В этом случае возможно использование PHP для создания клиентских [GUI](https://ru.wikipedia.org/wiki/GUI)-приложенийи решения административных задач в [операционных системах](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) [UNIX](https://ru.wikipedia.org/wiki/UNIX), [Linux](https://ru.wikipedia.org/wiki/Linux), [Microsoft Windows](https://ru.wikipedia.org/wiki/Microsoft_Windows), [Mac OS X](https://ru.wikipedia.org/wiki/Mac_OS_X) и [AmigaOS](https://ru.wikipedia.org/wiki/AmigaOS). Однако в таком качестве он не получил распространения, отдавая пальму первенства [Perl](https://ru.wikipedia.org/wiki/Perl)'у, [Python](https://ru.wikipedia.org/wiki/Python)'у и [VBScript](https://ru.wikipedia.org/wiki/Visual_Basic_Scripting_Edition)'у.

Начиная с версии PHP 5.4.0 в CLI SAPI появилась возможность запуска PHP как отдельного HTTP-сервера. Однако этот режим предназначен исключительно для разработки, так как запускает только один процесс интерпретатора и выполняет все запросы исключительно последовательно.

В силу популярности языка PHP и желания увеличить быстродействие основанных на нём [веб-приложений](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5), создано несколько альтернативных [компиляторов](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D0%B8%D0%BB%D1%8F%D1%82%D0%BE%D1%80), близких к PHP-языку. Так в феврале [2010 года](https://ru.wikipedia.org/wiki/2010_%D0%B3%D0%BE%D0%B4) компания [Facebook](https://ru.wikipedia.org/wiki/Facebook) открыла свой компилятор PHP — [HipHop](https://ru.wikipedia.org/wiki/HipHop_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) (HPHP, Hyper-PHP) генерирующий код на [Си](https://ru.wikipedia.org/wiki/C%2B%2B), с последующей компиляцией в машинный код с помощью [gcc](https://ru.wikipedia.org/wiki/Gcc). В июле 2013 года [российская](https://ru.wikipedia.org/wiki/%D0%A0%D0%BE%D1%81%D1%81%D0%B8%D1%8F) компания [ВКонтакте](https://ru.wikipedia.org/wiki/%D0%92%D0%9A%D0%BE%D0%BD%D1%82%D0%B0%D0%BA%D1%82%D0%B5) анонсировала похожий продукт - [KPHP](https://ru.wikipedia.org/wiki/KPHP).

**Javascript**

JavaScript является [объектно-ориентированным](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) языком, но используемое в языке [прототипирование](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%82%D0%BE%D1%82%D0%B8%D0%BF%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) обуславливает отличия в работе с объектами по сравнению с традиционными класс-ориентированными языками. Кроме того, JavaScript имеет ряд свойств, присущих [функциональным языкам](https://ru.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) - функции как [объекты первого класса](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82_%D0%BF%D0%B5%D1%80%D0%B2%D0%BE%D0%B3%D0%BE_%D0%BA%D0%BB%D0%B0%D1%81%D1%81%D0%B0), объекты как списки, [карринг](https://ru.wikipedia.org/wiki/%D0%9A%D0%B0%D1%80%D1%80%D0%B8%D0%BD%D0%B3), [анонимные функции](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%BE%D0%BD%D0%B8%D0%BC%D0%BD%D1%8B%D0%B5_%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%B8), [замыкания](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%BC%D1%8B%D0%BA%D0%B0%D0%BD%D0%B8%D0%B5_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) - что придаёт языку дополнительную гибкость.

Несмотря на схожий с Си синтаксис, JavaScript по сравнению с языком [Си](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8_(%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)) имеет коренные отличия:

* [объекты](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) с возможностью [интроспекции](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D1%80%D0%BE%D1%81%D0%BF%D0%B5%D0%BA%D1%86%D0%B8%D1%8F_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5));
* функции как [объекты первого класса](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82_%D0%BF%D0%B5%D1%80%D0%B2%D0%BE%D0%B3%D0%BE_%D0%BA%D0%BB%D0%B0%D1%81%D1%81%D0%B0);
* автоматическое [приведение типов](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5_%D1%82%D0%B8%D0%BF%D0%BE%D0%B2);
* автоматическая [сборка мусора](https://ru.wikipedia.org/wiki/%D0%A1%D0%B1%D0%BE%D1%80%D0%BA%D0%B0_%D0%BC%D1%83%D1%81%D0%BE%D1%80%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5));
* [анонимные функции](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%BE%D0%BD%D0%B8%D0%BC%D0%BD%D1%8B%D0%B5_%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%B8).

В языке отсутствуют такие полезные вещи, как:

* [модульная система](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D1%83%D0%BB%D1%8C_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)): JavaScript не предоставляет возможности управлять зависимостями и изоляцией областей видимости;
* [стандартная библиотека](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0): в частности, отсутствует [интерфейс программирования приложений](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F_%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B9) по работе с файловой системой, управлению потоками ввода-вывода, базовых типов для [бинарных данных](https://ru.wikipedia.org/wiki/%D0%94%D0%B2%D0%BE%D0%B8%D1%87%D0%BD%D1%8B%D0%B9_%D1%84%D0%B0%D0%B9%D0%BB);
* Стандартные [интерфейсы](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81) к [веб-серверам](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80) и [базам данных](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85);
* [система управления пакетами](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%BF%D0%B0%D0%BA%D0%B5%D1%82%D0%B0%D0%BC%D0%B8), которая бы отслеживала зависимости и автоматически устанавливала их.

[Синтаксис](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%81) языка JavaScript во многом напоминает синтаксис Си и [Java](https://ru.wikipedia.org/wiki/Java), [семантически](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D0%BC%D0%B0%D0%BD%D1%82%D0%B8%D0%BA%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) же язык гораздо ближе к [Self](https://ru.wikipedia.org/wiki/Self), [Smalltalk](https://ru.wikipedia.org/wiki/Smalltalk) или даже [Лиспу](https://ru.wikipedia.org/wiki/%D0%9B%D0%B8%D1%81%D0%BF).

В JavaScript:

* Все [идентификаторы](https://ru.wikipedia.org/wiki/%D0%98%D0%B4%D0%B5%D0%BD%D1%82%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%82%D0%BE%D1%80) регистрозависимы,
* в названиях переменных можно использовать [буквы](https://ru.wikipedia.org/wiki/%D0%91%D1%83%D0%BA%D0%B2%D0%B0), [подчёркивание](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%B4%D1%87%D1%91%D1%80%D0%BA%D0%B8%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5), [символ доллара](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D0%BC%D0%B2%D0%BE%D0%BB_%D0%B4%D0%BE%D0%BB%D0%BB%D0%B0%D1%80%D0%B0), [арабские цифры](https://ru.wikipedia.org/wiki/%D0%90%D1%80%D0%B0%D0%B1%D1%81%D0%BA%D0%B8%D0%B5_%D1%86%D0%B8%D1%84%D1%80%D1%8B),
* названия переменных не могут начинаться с цифры,
* для оформления однострочных [комментариев](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BC%D0%B5%D0%BD%D1%82%D0%B0%D1%80%D0%B8%D0%B8_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) используются //, многострочные и внутристрочные комментарии начинаются с /\* и заканчиваются \*/.

### Структура языка

Структурно JavaScript можно представить в виде объединения трёх чётко различимых друг от друга частей:

* ядро (ECMAScript),
* объектная модель браузера ([Browser Object Model или BOM](https://ru.wikipedia.org/w/index.php?title=Browser_Object_Model&action=edit&redlink=1" \o "Browser Object Model (страница отсутствует))),
* объектная модель документа ([Document Object Model или DOM](https://ru.wikipedia.org/wiki/Document_Object_Model" \o "Document Object Model)).

Если рассматривать JavaScript в отличных от браузера окружениях, то объектная модель браузера и объектная модель документа могут не поддерживаться.

Объектную модель документа иногда рассматривают как отдельную от JavaScript сущность, что согласуется с определением DOM как независимого от языка интерфейса документа. В противоположность этому ряд авторов находят BOM и DOM тесно взаимосвязанными.

ECMAScript не является браузерным языком и в нём не определяются методы ввода и вывода информации. Это, скорее, основа для построения скриптовых языков. Спецификация ECMAScript описывает типы данных, инструкции, ключевые и [зарезервированные](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D1%80%D0%B5%D0%B7%D0%B5%D1%80%D0%B2%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5_%D1%81%D0%BB%D0%BE%D0%B2%D0%BE) слова, [операторы](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%BE%D1%80_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), объекты, [регулярные выражения](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%B3%D1%83%D0%BB%D1%8F%D1%80%D0%BD%D1%8B%D0%B5_%D0%B2%D1%8B%D1%80%D0%B0%D0%B6%D0%B5%D0%BD%D0%B8%D1%8F), не ограничивая авторов производных языков в расширении их новыми составляющими.

Объектная модель браузера - браузер-специфичная часть языка, являющаяся прослойкой между ядром и объектной моделью документа. Основное предназначение объектной модели браузера - управление окнами браузера и обеспечение их взаимодействия. Каждое из окон браузера представляется объектом window, центральным объектом DOM. Объектная модель браузера на данный момент не стандартизирована, однако спецификация находится в разработке [WHATWG](https://ru.wikipedia.org/wiki/WHATWG) и [W3C](https://ru.wikipedia.org/wiki/W3C).

Помимо управления окнами, в рамках объектной модели браузера, браузерами обычно обеспечивается поддержка следующих сущностей:

* управление [фреймами](https://ru.wikipedia.org/wiki/%D0%A4%D1%80%D0%B5%D0%B9%D0%BC_(HTML)),
* поддержка задержки в исполнении кода и зацикливания с задержкой,
* системные диалоги,
* управление адресом открытой страницы,
* управление информацией о браузере,
* управление информацией о параметрах [монитора](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%BD%D0%B8%D1%82%D0%BE%D1%80_(%D1%83%D1%81%D1%82%D1%80%D0%BE%D0%B9%D1%81%D1%82%D0%B2%D0%BE)),
* ограниченное управление историей просмотра страниц,
* поддержка работы с [HTTP cookie](https://ru.wikipedia.org/wiki/HTTP_cookie).

Объектная модель документа - [интерфейс программирования приложений](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F_%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B9) для HTML и [XML](https://ru.wikipedia.org/wiki/XML)-документов. Согласно DOM, документ (например, веб-страница) может быть представлен в виде дерева объектов, обладающих рядом свойств, которые позволяют производить с ним различные манипуляции:

* генерация и добавление узлов,
* получение узлов,
* изменение узлов,
* изменение связей между узлами,
* удаление узлов.

Для добавления JavaScript-кода на страницу, можно использовать теги <script></script>, которые рекомендуется, но не обязательно, помещать внутри контейнера <head>. Контейнеров <script> в одном документе может быть сколько угодно. Атрибут «type='text/javascript'» указывать необязательно, данное значение используется по умолчанию.

Скрипт, выводящий [модальное окно](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE%D0%B5_%D0%BE%D0%BA%D0%BD%D0%BE) с классической надписью «Hello, World!» внутри браузера:

<script type="application/javascript">

alert('Hello, World!');

</script>

Элемент script, широко используемый для подключения к странице JavaScript, имеет несколько атрибутов.

* Необязательный атрибут type для указания [MIME-типа](https://ru.wikipedia.org/wiki/MIME) содержимого.
* необязательный атрибут src, принимающий в качестве значения адрес к файлу со скриптом.
* необязательный атрибут charset, используемый вместе с src для указания используемой кодировки внешнего файла.
* необязательный атрибут defer указывает, что получение скрипта происходит асинхронно, но выполнение следует отложить до тех пор, пока страница не будет загружена целиком.
* необязательный атрибут async указывает, что получение скрипта происходит асинхронно, а выполнение будет произведено сразу по завершению скачивания. Очерёдность выполнения скриптов не гарантируется.

При этом атрибут language (language="JavaScript"), несмотря на его активное использование (в 2008 году этот атрибут был наиболее часто используемым у тега <script>), относится к не рекомендуемым (deprecated), отсутствует в [DTD](https://ru.wikipedia.org/wiki/DTD), поэтому считается некорректным.

**Применение**

### Веб-приложения

JavaScript используется в клиентской части веб-приложений: клиент-серверных программ, в котором клиентом является браузер, а сервером - веб-сервер, имеющих распределённую между сервером и клиентом логику. Обмен информацией в веб-приложениях происходит по сети. Одним из преимуществ такого подхода является тот факт, что клиенты не зависят от конкретной операционной системы пользователя, поэтому веб-приложения являются кроссплатформенными сервисами.

* **AJAX**

JavaScript используется в [AJAX](https://ru.wikipedia.org/wiki/AJAX), популярном подходе к построению интерактивных пользовательских интерфейсов веб-приложений, заключающемся в «фоновом» асинхронном обмене данными браузера с веб-сервером. В результате, при обновлении данных веб-страница не перезагружается полностью и интерфейс веб-приложения становится быстрее, чем это происходит при традиционном подходе (без применения AJAX).

* **Comet**

Comet - широкое понятие, описывающее механизм работы веб-приложений, использующих постоянные HTTP-соединения, что позволяет веб-серверу отправлять данные браузеру без дополнительного запроса со стороны браузера. Для таких приложений используются технологии, непосредственно поддерживаемые браузерами. В частности, в них широко используется JavaScript.

* **Браузерные операционные системы**

JavaScript широко используется в [браузерных операционных системах](https://ru.wikipedia.org/wiki/WebOS). Так, например, исходный код [IndraDesktop WebOS](https://ru.wikipedia.org/w/index.php?title=IndraDesktop_WebOS&action=edit&redlink=1) на 75% состоит из JavaScript, код браузерной операционной системы [IntOS](https://ru.wikipedia.org/w/index.php?title=IntOS&action=edit&redlink=1) - на 70%. Доля JavaScript в исходном коде [eyeOS](https://ru.wikipedia.org/wiki/EyeOS) - 5%, однако, и в рамках этой операционной системы JavaScript играет важную роль, участвуя, в визуализации на клиенте и, являясь, необходимым механизмом для коммуницирования клиента и сервера.

### Букмарклеты

JavaScript используется для создания небольших программ, размещаемых в закладки браузера. При этом используются URL-адреса со спецификатором javascript:.

### Пользовательские скрипты в браузере

Пользовательские скрипты в браузере - это программы, написанные на JavaScript, выполняемые в браузере пользователя при загрузке страницы. Они позволяют автоматически заполнять формы, переформатировать страницы, скрывать нежелательное содержимое и встраивать желательное для отображения содержимое, изменять поведение клиентской части веб-приложений, добавлять элементы управления на страницу и т. д.

Для управления пользовательскими скриптами в [Mozilla Firefox](https://ru.wikipedia.org/wiki/Mozilla_Firefox) используется расширение [Greasemonkey](https://ru.wikipedia.org/wiki/Greasemonkey); [Opera](https://ru.wikipedia.org/wiki/Opera) и [Google Chrome](https://ru.wikipedia.org/wiki/Google_Chrome)предоставляют средства поддержки пользовательских скриптов и возможности для выполнения ряда скриптов Greasemonkey.

### Серверные приложения

Приложения, написанные на JavaScript, могут исполняться на серверах, использующих [Java](https://ru.wikipedia.org/wiki/Java) 6 и более поздних версий. Это обстоятельство используется для построения серверных приложений, позволяющих обрабатывать JavaScript на стороне сервера.

Помимо Java 6, существует ряд платформ, использующих существующие движки (интерпретаторы) JavaScript для исполнения серверных приложений. (Как правило, речь идёт о повторном использовании движков, ранее созданных для исполнения кода JavaScript в браузерах WWW)

### Мобильные приложения

Перевод мобильных устройств [Palm](https://ru.wikipedia.org/wiki/Palm_(%D0%9A%D0%9F%D0%9A)) на использование [Palm webOS](https://ru.wikipedia.org/wiki/Palm_webOS) в качестве операционной системы с [Mojo SDK](https://ru.wikipedia.org/w/index.php?title=Mojo_SDK&action=edit&redlink=1) в качестве [комплекта средств разработки](https://ru.wikipedia.org/wiki/SDK)позволяет использовать JavaScript в качестве языка разработки мобильных приложений.

### Виджеты

Виджет - вспомогательная мини-программа, графический модуль которой размещается в рабочем пространстве соответствующей [родительской программы](https://ru.wikipedia.org/w/index.php?title=%D0%94%D0%B2%D0%B8%D0%B6%D0%BE%D0%BA_%D0%B2%D0%B8%D0%B4%D0%B6%D0%B5%D1%82%D0%BE%D0%B2&action=edit&redlink=1) ([англ.](https://en.wikipedia.org/wiki/Widget_engine)), служащая для украшения рабочего пространства, развлечения, решения отдельных рабочих задач или быстрого получения информации из интернета без помощи веб-браузера. JavaScript используется как для реализации виджетов, так и для реализации движков виджетов. В частности, при помощи JavaScript реализованы [Apple Dashboard](https://ru.wikipedia.org/wiki/Apple_Dashboard), [Microsoft Gadgets](https://ru.wikipedia.org/w/index.php?title=Microsoft_Gadgets&action=edit&redlink=1), [Yahoo!\_Widgets](https://ru.wikipedia.org/w/index.php?title=Yahoo!_Widgets&action=edit&redlink=1), [Google Gadgets](https://ru.wikipedia.org/w/index.php?title=Google_Gadgets&action=edit&redlink=1), [Klipfolio Dashboard](https://ru.wikipedia.org/w/index.php?title=Klipfolio_Dashboard&action=edit&redlink=1).

### Прикладное программное обеспечение

JavaScript используется для написания [прикладного ПО](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%BA%D0%BB%D0%B0%D0%B4%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5). Например, **16,4 %** исходного кода [Mozilla Firefox](https://ru.wikipedia.org/wiki/Mozilla_Firefox) написано на JavaScript.

[Google Chrome OS](https://ru.wikipedia.org/wiki/Google_Chrome_OS) в качестве прикладного ПО использует [веб-приложения](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5).

В окружении рабочего стола [GNOME](https://ru.wikipedia.org/wiki/GNOME) имеется возможность создавать на JavaScript программы, оперирующие с библиотеками GNOME при помощи [Gjs](https://ru.wikipedia.org/w/index.php?title=Gjs&action=edit&redlink=1),[Seed](https://ru.wikipedia.org/w/index.php?title=Seed&action=edit&redlink=1).

### Манипуляция объектами приложений

JavaScript также находит применение в качестве скриптового языка доступа к объектам приложений. Платформа Mozilla ([XUL](https://ru.wikipedia.org/wiki/XUL)/[Gecko](https://ru.wikipedia.org/wiki/Gecko_(%D0%B4%D0%B2%D0%B8%D0%B6%D0%BE%D0%BA))) использует JavaScript. Среди сторонних продуктов, например, [Java](https://ru.wikipedia.org/wiki/Sun_Java_Runtime_Environment), начиная с версии 6, содержит встроенный интерпретатор JavaScript на базе [Rhino](https://ru.wikipedia.org/wiki/Rhino). Сценарии JavaScript поддерживаются в таких приложениях Adobe, как [Adobe Photoshop](https://ru.wikipedia.org/wiki/Adobe_Photoshop), [Adobe Dreamweaver](https://ru.wikipedia.org/wiki/Adobe_Dreamweaver), [Adobe Illustrator](https://ru.wikipedia.org/wiki/Adobe_Illustrator) и [Adobe InDesign](https://ru.wikipedia.org/wiki/Adobe_InDesign).

### Офисные приложения

JavaScript используется в [офисных приложениях](https://ru.wikipedia.org/wiki/%D0%9E%D1%84%D0%B8%D1%81%D0%BD%D1%8B%D0%B9_%D0%BF%D0%B0%D0%BA%D0%B5%D1%82) для автоматизации рутинных действий, написания [макросов](https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D0%BA%D1%80%D0%BE%D1%81), организации доступа со стороны веб-служб.