1. **Среда для установки** 
   1. **PHPStorm J:\Программирование Основы\Прога\JetBrains PhpStorm 2018.1.4 Build #PS-181.5087.24 Eng**
   2. **GitHub Desktop**
   3. **Notepad++** **J:\Программирование Основы\Прога\NOTEPAD ++**
   4. **OpenSever J:\Программирование Основы\Прога\Open Server phpMyAdmin (ver.5.3.0)**
   5. **Git-2.21.0-64-bit.exe J:\Программирование Основы\Прога\Git**
   6. **Android Studio**
   7. **Visual Studio** [**Community 2019**](https://visualstudio.microsoft.com/ru/thank-you-downloading-visual-studio/?sku=Community&rel=16)
2. [**ОС операционная система**](#ОС_операционная_система)
   1. [**Оболо́чка операцио́нной систе́мы**](#Оболочка_ОС)
3. [**Windows Загрузчик меню Загрузки**](#ОС_операционная_система)
   1. [**VHD Windows**](#VHD_windows)
   2. [**Среда предустановки Windows (англ. Windows Preinstallation Environment, WinPE**](#Windows_PE)
4. [**Переменная среды**](#Переменные_среды)
5. [**C#(Си шарп) C++**](#C_C_plusplus)
6. [**Visual Studio**](#Visual_Studio)
7. **[Visual Studio (VSCode)](#Visual_Studio_Code)**
8. [**Microsoft Expression Blend**](#Blend_Microsoft)
9. [**GNU**](#GNU)
10. [**Linux**](#Linux)
    1. [**Linux\_mint**](#Linux_mint)
11. [**FreeBSD**](#FreeBSD)
12. [**НАТИВНЫЕ И КРОССПЛАТФОРМЕННЫЕ ПРИЛОЖЕНИЯ**](#НАТИВНЫЕ_КРОССПЛАТФОРМЕННЫЕ_ПРИЛОЖЕНИЯ)
13. [**WEB верстка**](#WEB_верстка)
14. [**Виртуальные машины**](#Виртуальные_машины)
    1. [**Аппаратная виртуализация**](#аппаратная_виртуализация)
    2. [**HTML**](#HTML) **(от англ. HyperText Markup Language — «язык разметки гипертекста»)**
       1. [**XHTML (англ. extensible hypertext markup language — расширяемый язык гипертекстовой разметки)**](#XHTML)
       2. [**Мнемоника**](#мнемоника)**(сущность)HTML**
       3. **Шрифты**
       4. [**DOM** модель](#DOM_модель)
    3. [**CSS**](#CSS) **(англ. Cascading Style Sheets — каскадные таблицы стилей)**
       1. CSS препроцессор (CSS preprocessor) - это программа, которая имеет свой собственный синтаксис (syntax), но может сгенерировать из него CSS код
          1. [SASS](http://sass-lang.com/)
          2. [LESS](http://lesscss.org/)
          3. [Stylus](http://stylus-lang.com/)
          4. [PostCSS](http://postcss.org/)
    4. [**Javascript**](#Javascript)
       1. [**jQuery**](#jQuery)
    5. [**XML** (eXtensible Markup Language) — расширяемый язык разметки](#XML)
    6. [**SGML**](#SGML) **(**[**англ.**](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA)**Standard Generalized Markup Language — стандартный обобщённый язык разметки)**
15. [**WEB серверная часть**](#Серверная_часть)
    1. [**Beget.ru сервер подключения**](#BEGET_RU)
16. [**Протоколы**](#Протоколы)
    1. [**SSH (Secure Shell — «безопасная оболочка»)**](#SSH) **— сетевой протокол прикладного уровня**
    2. [**Отличие HTTP от HTTPS**](#HTTP_HTTPS)
17. [**СУБД**](#СУБД)
18. [**Open Server Panel**](#Open_server)
19. [**.htaccess**](#htaccess)
    1. [**SQL**](#SQL)
    2. **[MySQL](#MySQL) (structured query language — «язык структурированных запросов»)**
       1. [**Предикаторы**](#Предикаторы)
    3. [**PHP (Hypertext Preprocessor — «PHP: препроцессор гипертекста**](#PHP)**»)**
       1. [**Исключение — это объект встроенного в PHP класса Exception**](#Исключение_PHP)
       2. [**Composer(getcomposer.org**](#Composer)**)**
    4. [**PDO PHP**](#PDO) **(PHP Data Objects)**
    5. [**Git система контроля версий**](#Git)
       1. **https://git-scm.com/download/gui/windows**
       2. [**Vi или Vim редактор**](#Vi_Unix_редактор)
    6. **[GitHub](#GitHub) хранение внешних репозиториев https://desktop.github.com**
       1. **Aton – Редактор GitHub**
    7. [**ООП**](#ООП)
       1. [**Паттерны**](#Паттерны)
       2. [**Фреймворк**](#Фреймворк)
          1. [**фрэймворк Bootstrap**](#Bootstrap)
       3. **Наследование**
       4. [**Сущность**](#Сущность)
       5. [**Полиморфизм**](#Полиморфизм)
       6. [**Инкапсуляция**](#Инкапсуляция)
       7. [**Наследование**](#Наследование)
20. [**PSR (PHP standart recommendation) стандартов**](#PSR)
21. [**NOTEPAD++ Особенности настройки**](#NOTEPADplusplus)
22. [**КОНСОЛИ Windows**](#Консоли_Windows)
    1. [**CMD**](#CMDer)**er**
    2. **CMD\_Windows**
    3. [**Windows powershell**](#Power_shell_windows)
    4. [**Bash**](#Bash)
23. [**Total commander**](#Total_comander)
24. **[БРАУЗЕРЫ](#браузеры)**
    1. [**Плагины**](#браузеры)
25. **Cайты для имитации просмотра кода на различных мобильных устройствах**
    1. [**http://ami.responsivedesign.is**](http://ami.responsivedesign.is)
    2. [**www.responsinator.com**](http://www.responsinator.com)
26. [**СЕТИ HOST VPN**](#Сети_HOST_VPN)
27. [**ПРЕПРОЦЕССОР**](#Препроцеесор)
28. [**Задачи с ответами**](#Задачи_с_ответами)
    1. [**для программирования PHP**](#PHP_ЗАДАЧИ)
29. [**Программы для Верстки**](#Программы_для_верстки)
    1. [**PHPStorm**](#PHPStorm)
       1. PhpStorm 2018.1.4 Build #PS-181.5087.24, built on May 18, 2018 уст.версия
          1. Настройка авт.подсветки EMMET тегов Ctrl-Alt-S (Setting) -> Editor->Emmet->HTML->Enable abbreviation preview
       2. **IntelliJ IDEA** — [интегрированная среда разработки программного обеспечения](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D0%B3%D1%80%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D1%80%D0%B5%D0%B4%D0%B0_%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8) для многих языков программирования, в частности [Java](https://ru.wikipedia.org/wiki/Java), [JavaScript](https://ru.wikipedia.org/wiki/JavaScript), [Python](https://ru.wikipedia.org/wiki/Python), разработанная компанией [JetBrains](https://ru.wikipedia.org/wiki/JetBrains).
       3. **Emmet плагин.** **Базовый синтаксис плагина EMMET** [**https://dwstroy.ru/stail/plaginy-rasshireniya/emmet-shpargalka/**](https://dwstroy.ru/stail/plaginy-rasshireniya/emmet-shpargalka/)

**https://docs.emmet.io**

* + 1. [**Сниппет**](#Snippet)
    2. **Плагин** 
       1. **Live Edit предназначен для интерактивного отслеживания измения в программе (только для Chrom) урок Live Edit в PHPStorm установка и работаместо раположение \Программирование Основы\Уроки\! уроки общие изучения\PHPStorm настройка**
  1. **Java**
  2. [**ПОДКЛЮЧЕНИЯ BEGET.RU**](#BEGET_RU)

1. [**Уроки HTML 5 и CSS**](#Уроки_HTML5_CSS_Js)
2. [**Регулярные выражения**](#Регулярные_выражения)
3. [**Консоль открыть**](#консоль)
4. [**XSS атаки ( уязвимость)**](#XSS_атаки_уязвимость)
5. **Статьи**
   1. [Как стать начинающим верстальщиком](#Как_стать_верстальщиком)
6. [**Контакты WEB разработчиков**](#Контакты_WEB_разработчиков)
7. [**Термины**](#термины)
   1. [API(application programming interface)](#API)
   2. [Md5 хеширование](#MD5)
   3. [Парсинг (парсер](#Парсинг))
   4. [Тик](#Тик)
   5. [Плейсхолдеры](#Плейсхолдеры)
   6. [MIME-типы файлов](#MIME)
   7. [GUI , англ.graphical user interface](#GUI)
   8. [Терна́рная усло́вная опера́ция](#Тернарный_оператор)
   9. [**Псевдотипы и переменные, используемые в документации PHP**](#Псевдотипы)
      1. $statement
      2. mixed
      3. number
      4. callback
      5. array|object
      6. [$void](#void)
   10. [Рефа́кторинг](#рефакторинг)
   11. [Псевдоклассы](#псевдоклассы)
   12. [**Drag-and-drop**](#Drag_and_Drop)
8. [Сервисы Фриланс](#Сервисы_Фриланс)
9. [написание программ для андроид](#Proga_android)
10. [СЕТИ VPN DNS DDNS](#СЕТИ_VPN_DDNS_SSB)
11. [**Вопросы**](#Вопросы)

Веб-разработка — процесс создания веб-сайта или веб-приложения. Основными этапами процесса являются веб-дизайн, вёрстка страниц, программирование для веб на стороне клиента и сервера, а также конфигурирование веб-сервера.

**Этапы WEB разработки** - <https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B0>

**Вёрстка веб-страниц** -создание структуры html-кода, размещающего элементы веб-страницы (изображения, текст и т. д.) в окне браузера, согласно разработанному макету[1], таким образом, чтобы элементы дизайна выглядели аналогично макету (https://ru.wikipedia.org/wiki/%D0%92%D1%91%D1%80%D1%81%D1%82%D0%BA%D0%B0\_%D0%B2%D0%B5%D0%B1-%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%86)

WEB менеджер, WEB верстка

Серверные разработки

Программирование мобильных приложений

**Программы для Верстки**

**Notepad++**

**SourceForge.net —** один из самых больших в мире веб-сайтов для разработчиков открытого программного обеспечения, который обслуживается SourceForge, Inc. и использует проприетарную систему SourceForge. Недоступен на территории Крыма

**Sublime Text** <https://www.sublimetext.com> кроссплатформенный проприетарный текстовый редактор. Поддерживает плагины на языке программирования Python.

Разработчик позволяет бесплатно и без ограничений ознакомиться с продуктом, однако программа уведомляет о необходимости приобретения лицензии.

**PhpStorm** [**https://www.jetbrains.com/phpstorm/?frommenu**](https://www.jetbrains.com/phpstorm/?frommenu)

**phpDesinger7 аналог Notepad++ (платная)**

**Dreamweaver** (*Дримвивер*) — визуальный [HTML-редактор](https://ru.wikipedia.org/wiki/HTML-%D1%80%D0%B5%D0%B4%D0%B0%D0%BA%D1%82%D0%BE%D1%80) компании [Adobe](https://ru.wikipedia.org/wiki/Adobe). Не использовать

**WeBuilder 2018 v15.2.0.204 МОРЕВ САША <https://rsload.net/soft/editor/9572-webuilder.html>**

-------------

Контент-менеджер — специалист по созданию, распространению и курированию контента, редактор сайтов. В обязанности контент-менеджера входит наполнение сайта текстовой, графической и другими видами информации, полезной и удобной для восприятия выбранной целевой группой (контентом).

Вёрстка веб-страниц - создание структуры html-кода, размещающего элементы веб-страницы (изображения, текст и т. д.) в окне браузера, согласно разработанному макету[1], таким образом, чтобы элементы дизайна выглядели аналогично макету[2].Вёрстка веб-страниц отличается от полиграфической тем, что необходимо учитывать разницу отображения элементов в различных браузерах и разницу в размерах рабочего пространства устройств[1].Процесс сложен и имеет творческую основу, ни один из способов не является каноничным и принятым как основа[1]. Все подходы к вёрстке имеют как преимущества, так и недостатки..

# ТЕРМИНЫ

**API (от англ. Application Program Interface)** – это интерфейс взаимодействия между сайтом клиента и сервером.

**CSS (/siːɛsɛs/ англ. Cascading Style Sheets** — каскадные таблицы стилей)

Спецификация языка HTML5 **<https://www.w3schools.com>**

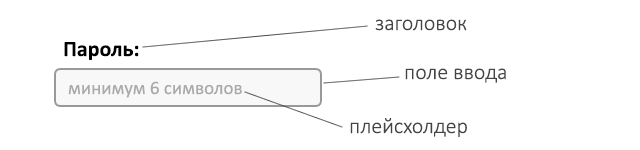
# MD5 (англ. Message Digest 5) — 128-битный алгоритм хеширования, разработанный профессором Рональдом Л. Ривестом из Массачусетского технологического института (Massachusetts Institute of Technology, MIT) в 1991 году. Предназначен для создания «отпечатков» или дайджестов сообщения произвольной длины и последующей проверки их подлинности. Широко применялся для проверки целостности информации и хранения хешей паролей.

<https://ru.wikipedia.org/wiki/MD5>

* **Парсинг (парсер)** Полное или частичное копирование материалов сайта с последующим размещением этих материалов на своих ресурсах. Например, для использования на [сателлитах](https://myblaze.ru/chto-takoe-setka-saytov/). При этом текст может быть предварительно пропущен через синонимайзер или обработан рерайтером для повышения уникальности.
* Для интерпритации JS используется [http://esprima.org/demo/parse.html#](http://esprima.org/demo/parse.html)

**Тик - это событие**, которое случается каждые N низкоуровневых операций, выполненных парсером внутри блока declare. Значение N задается, используя ticks=N внутри секции directive блока declare.

**Плейсхолдеры**

Вспомогательный текст внутри полей ввода

**Плейсхолдеры** (от англ. placeholders — в интернет-маркетинге: указатель места заполнения или подсказка, расположенная внутри каждого поля подписной формы): это технология, которая, по замыслу авторов должна облегчить процесс заполнения лид-формы и способствовать повышению коэффициента конверсии.

Но как свидетельствуют многочисленные исследования, плейсхолдеры приносят больше вреда, чем пользы.

<https://habr.com/ru/post/235117/>

**MIME (Multipurpose Internet Mail Extension, Многоцелевые расширения почты Интернета)**

**MIME (Multipurpose Internet Mail Extension, Многоцелевые расширения почты Интернета)** — спецификация для передачи по сети файлов различного типа: изображений, музыки, текстов, видео, архивов и др. Указание MIME-типа используется в HTML обычно при передаче данных форм и вставки на страницу различных объектов.

<https://ru.wikipedia.org/wiki/MIME>

Некоторые популярные MIME-типы и расширения файлов с ними связанных приведены в табл. 1.

|  |  |
| --- | --- |
| *Табл. 1. MIME-типы* | |
| **Расширение файла** | **Тип данных** |
| Ai | application/postscript |
| Aif | audio/aiff |
| Aif | audio/x-aiff |
| Aiff | audio/aiff |
| Aiff | audio/x-aiff |
| Ani | application/x-navi-animation |
| Book | application/book |
| C | text/x-c |
| c++ | text/plain |

**Терна́рная усло́вная опера́ция**

**Терна́рная усло́вная опера́ция** (от лат. ternarius — «тройной») (обычно записывается как ?:) — во многих языках программирования операция, возвращающая свой второй или третий операнд в зависимости от значения логического выражения, заданного первым операндом. Аналогом тернарной условной операции в математической логике и булевой алгебре является условная дизъюнкция, которая записывается в виде [p, q, r] и реализует алгоритм: «Если p, то q, иначе r», что можно переписать как «q или r, в зависимости от p или не p».

**Псевдотипы и переменные, используемые в документации PHP**

**http://php.net/manual/ru/language.pseudo-types.php#language.types.mixed**

**mixed ¶**

**mixed** говорит о том, что параметр может принимать много (но необязательно все) типов.

Например, функция gettype() принимает все типы PHP, тогда как str\_replace() принимает только типы string и array.

**number** ¶

**numbe**r говорит о том, что параметр может быть либо integer, либо float.

**callback** ¶

Псевдотип **callback** использовался в этой документации до того, как был введен тип callable в PHP 5.4. Он означает в точности то же самое.

**array|object** ¶

**array|object** указывает, что параметр может быть как массивом array, так и объектом object.

**void**

**void** — void в качестве типа результата означает, что возвращенное значение бесполезно. void в списке параметров означает, что функция не принимает параметров. Начиная с PHP 7.1, void разрешается использовать в качестве подсказки возвращаемого типа функции. тип, спецификатор типа и ключевое слово в языках программирования С, C++, Java, C#, Objective-C, D, ActionScript и PHP.

**$...**

**$...** в прототипах функции означает and so on (и так далее). Это имя переменной используется, когда функция может принимать бесконечное количество параметров.

**Рефа́кторинг**

или перепроектирование кода, переработка кода, равносильное преобразование алгоритмов — процесс изменения внутренней структуры программы, не затрагивающий её внешнего поведения и имеющий целью облегчить понимание её работы.

**Сервисы Фриланс**

<https://kwork.ru> password mir3113500 login [mir-hai@mail.ru](mailto:mir-hai@mail.ru) имя mir-hai

<http://kworks.ru> имя John3111 login [mir-hai@mail.ru](mailto:mir-hai@mail.ru) Password alfa31113500

**Уроки HTML 5 и CSS и JS**

Прога для автоматического просмотра верстки сайта

1) http://cssdeck.com/labs

2) <http://jsfiddle.net>

**https://www.youtube.com/playlist?list=PLY4rE9dstrJyeZlPWoKJr1xKVVnG4w-Hc**

**Css https://www.youtube.com/playlist?list=PL026CCEB5125879C2**

**Javascript https://www.youtube.com/playlist?list=PL363QX7S8MfSxcHzvkNEqMYbOyhLeWwem**

**Git** [**https://www.youtube.com/playlist?list=PLY4rE9dstrJyTdVJpv7FibSaXB4BHPInb**](https://www.youtube.com/playlist?list=PLY4rE9dstrJyTdVJpv7FibSaXB4BHPInb)

JS <https://www.youtube.com/watch?v=ch7eVYUvb_w&list=PLVfMKQXDAhGUaEtJ_fOUz0F7TJtidE7Qw>

[**https://www.youtube.com/watch?v=5CPTpFit3hg**](https://www.youtube.com/watch?v=5CPTpFit3hg)

[**https://www.youtube.com/watch?v=nQL5rAB11i0**](https://www.youtube.com/watch?v=nQL5rAB11i0)

[**https://www.youtube.com/watch?v=LVB8sFslfb0**](https://www.youtube.com/watch?v=LVB8sFslfb0)

[**https://www.youtube.com/watch?v=QyjRcJVTEhk**](https://www.youtube.com/watch?v=QyjRcJVTEhk)

**JS <https://www.youtube.com/watch?v=ch7eVYUvb_w&list=PLVfMKQXDAhGUaEtJ_fOUz0F7TJtidE7Qw> уроки**

**UTF-8 урок** [**https://yadi.sk/d/kpHk33dOVp77j**](https://yadi.sk/d/kpHk33dOVp77j)

**Регулярные выражения**

**https://www.youtube.com/watch?v=VCgAvj9meHg**

Георгий (Гоша) Дударь

Дополнительное описание, домашние задания и многое другое можно найти на сайте itProger: https://itproger.com/course/html

✔ Основной сайт: https://itproger.com/

Построение CSS - <https://ru.wikipedia.org/wiki/CSS>

1. **ОС операционная система** 
   1. **Оболочка ОС**
      1. Оболо́чка операцио́нной систе́мы (от англ. shell «оболочка») — интерпретатор команд операционной системы, обеспечивающий интерфейс для взаимодействия пользователя с функциями системы. В общем случае различают оболочки с двумя типами интерфейса для взаимодействия с пользователем: текстовый пользовательский интерфейс (TUI) и графический пользовательский интерфейс (GUI). <https://ru.wikipedia.org/wiki/Оболочка_операционной_системы>
      2. Для обеспечения интерфейса командной строки в операционных системах часто используются командные интерпретаторы, которые могут представлять собой самостоятельные языки программирования с собственным синтаксисом и отличительными функциональными возможностями.
      3. В операционные системы MS-DOS и Windows 9x включён командный интерпретатор command.com, в Windows NT включён cmd.exe, начиная с Windows XP (пакет обновления 2) доступен PowerShell, который является встроенным компонентом ОС начиная с Windows 7 и Windows 2008 Server.
      4. В UNIX-подобных системах у пользователя есть возможность менять командный интерпретатор, используемый по умолчанию. Из командных оболочек UNIX наиболее популярны bash, csh, ksh, zsh.
   2. **Сам по себе "shell" это интерпритатор команд**
      1. **cmd, powershell, bash, shell, terminal это оболочки интепритатора.**
2. **Windows Загрузчик меню Загрузки**
   1. **Меню загрузки**
      1. EasyBCD отдельная программ ( есть бесплатная версия) [**https://ru.wikipedia.org/wiki/EasyBCD**](https://ru.wikipedia.org/wiki/EasyBCD) сайт загрузки [**https://neosmart.net/EasyBCD/**](https://neosmart.net/EasyBCD/) **20.04.12\_EasyBCD\_восстановление загрузки.bcd** в **E:\Dropbox\my файл восстановления**
      2. Редактирование меню загрузки с помощью **bcdedit.exe через терминал bcdedit.exe /? Справка по команде**
      3. «Загрузка операционной системы» можно так: Пуск – «Панель управления\Все элементы панели управления\Система»: «Панель управления\Система и безопасность\Система». «Дополнительные параметры системы», на вкладку «Дополнительно»
      4. Нажимаем «Win+R» для вызова окна «Выполнить». Вписываем в строку «msconfig»
   2. **VHD Windows**
      1. Virtual Hard Disk (VHD) — формат файла, содержащий полную структуру и содержание, сходные с жёстким диском. Используется для хранения виртуальных операционных систем, программ и других файлов в одном файле-образе, который можно открыть разными программами виртуализации или виртуальными машинами.[

Изначально формат был создан компанией Connectix и позднее куплен Microsoft вместе с программой виртуализации Virtual PC. C июня 2005 Microsoft сделала спецификацию формата VHD доступной третьим фирмам в рамках Microsoft Open Specification Promise.

<https://ru.wikipedia.org/wiki/VHD>

* 1. **Среда предустановки Windows (англ. Windows Preinstallation Environment, WinPE)**

Среда предустановки Windows (англ. Windows Preinstallation Environment, WinPE) — облегчённые версии операционных систем Microsoft Windows, позволяющие загружаться с компакт-дисков, usb (flash или HDD) или же по сети. реда предустановки Windows изначально предназначалась для использования в качестве предустановочной платформы для запуска Microsoft Windows, как замена [DOS](https://ru.wikipedia.org/wiki/DOS). Позднее крупные компании стали использовать её для предустановки Windows на большое число компьютеров, а также для восстановления работоспособности операционной системы. Продукт может быть использован разработчиками для тестирования.

Изначально для создания образов Среды предустановки Windows необходимо было обладать подпиской SA (Software Assurance), OEM или ISV-лицензией. В версии 2.0 это ограничение было снято

<https://ru.wikipedia.org/wiki/%D0%A1%D1%80%D0%B5%D0%B4%D0%B0_%D0%BF%D1%80%D0%B5%D0%B4%D1%83%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BA%D0%B8_Windows#Windows_PE_10>

1. **Переменная среды**

**Переменная среды́** (англ. environment variable) — текстовая переменная операционной системы, хранящая какую-либо информацию — например, данные о настройках системы.

**Переменные среды́ UNIX**

Переменные среды́ устанавливаются пользователем или сценариями оболочки. Начальный набор переменных задаётся стартовыми сценариями операционной системы и сценариями, запускаемыми при регистрации пользователя в системе, в частности, /etc/profile и ~/.bash\_profile.

В сценариях обычно используются объявления вида ИМЯ\_ПЕРЕМЕННОЙ="значение переменной", но конкретный синтаксис зависит от используемого интерпретатора.

**Переменные среды́ Windows**

Переменные среды́ задаются в реестре Windows и программным обеспечением. Они могут использоваться в командной строке и в адресной строке проводника

Переменные среды́ Windows делятся на две категории:

Переменные среды́ пользователя — указывают путь до пользовательских каталогов.

Системные переменные — хранят данные о некоторых каталогах операционной системы и конфигурации компьютера.

Обращение к некоторой переменной среды́ возможно таким образом: **%ИМЯ\_ПЕРЕМЕННОЙ%**

Вывод содержимого переменной — echo %VAR%.

Вывод первых четырёх символов — echo %VAR:~0,4%.

Вывод символов с третьего по шестой — echo %VAR:~2,4%

Вывод последних четырёх символов — echo %VAR:~-4%

[языки программирования.xlsx](file:///H:\Dropbox\Личная%20моя\domains\языки%20программирования.xlsx)

**Виртуальные машины**

1 Oracle Virtualbox Некоторые настройки ВМ и гостевой системы

Виртуальная машина [Oracle Virtualbox](https://www.virtualbox.org/), пожалуй, наиболее известна и популярна у домашних пользователей ПК. Она русифицирована, бесплатна, поддерживает [все основные операционные системы](https://www.virtualbox.org/wiki/Guest_OSes), включая Android, и довольно проста в применении. Сделать минимально необходимые настройки и установить в нее операционную систему сможет даже слабо подготовленный юзер, если имеет под рукой инструкцию по установке выбранной ОС.

<https://www.virtualbox.org>

2 VMware Workstation Pro Некоторые опции виртуальной машины VMware

Скачать можно с torrent

https://compconfig.ru/software/virtualnye-mashiny-dlya-windows.html#VMware\_Workstation\_Pro

3 Microsoft Hyper-V Некоторые опции виртуальных машин Hyper-V

**Для функционирования Hyper-V компьютер должен соответствовать следующим требованиям:**

**Процессор должен быть 64-разрядным. Hyper-V доступен в 64-разрядных версиях Windows.**

**Компьютер должен поддерживать аппаратную виртуализацию. Это возможно в процессорах, поддерживающих параметр виртуализации, а именно Intel Virtualization Technology (Intel VT) и AMD Virtualization (AMD-V). Эта функция должна быть включена в BIOS.**

**Нажать на сочетание клавиш «WIN+R«;**

1. **В строке «Выполнить» вписать команду к appwiz.cpl**

**Аппаратная виртуализация — виртуализация с поддержкой специальной процессорной архитектуры. В отличие от программной виртуализации, с помощью данной техники возможно использование изолированных гостевых систем, управляемых гипервизором напрямую.**

В Intel VT (Intel Virtualization Technology) реализована виртуализация режима реальной адресации (режим совместимости с 8086). Соответствующая аппаратная виртуализация ввода-вывода — VT-d (кодовое название — Vanderpool). Часто обозначается аббревиатурой **VMX (Virtual Machine eXtension).**

**Dell Intel Core i5-2450M CPU @ 2.50GHz**

**VT-x (Virtualization technology)-да**

**виртуализации Intel® для направленного ввода/вывода (VT-d) - нет**

**Homecomputer Intel® Core™2 Duo E8400**

**Технология виртуализации Intel® (VT-x) ‡**

Технология Intel® Virtualization для направленного ввода/вывода (VT-x) позволяет одной аппаратной платформе функционировать в качестве нескольких «виртуальных» платформ. Технология улучшает возможности управления, снижая время простоев и поддерживая продуктивность работы за счет выделения отдельных разделов для вычислительных операций.

**Технология виртуализации Intel® для направленного ввода/вывода (VT-d) ‡**

Технология Intel® Virtualization Technology для направленного ввода/вывода дополняет поддержку виртуализации в процессорах на базе архитектуры IA-32 (VT-x) и в процессорах Itanium® (VT-i) функциями виртуализации устройств ввода/вывода. Технология Intel® Virtualization для направленного ввода/вывода помогает пользователям увеличить безопасность и надежность систем, а также повысить производительность устройств ввода/вывода в виртуальных средах.

**Панель управления->Панель управления\Программы\компоненты Windows\Hyper-V**

**WEB вертка**

1. **Верстка** – это процесс преобразования дизайна в Web Страницу
2. **Главная задача**, что бы страница одинаково хорошо смотрелась в различных броузерах и на мониторах с различным расширением
3. **Критерии верстки –** адаптация под разные браузеры и разные разрешения экранов
   1. **Существуют типы верстка:**
      1. **адаптивная верстка**
         1. CSS @media
         2. HTML адаптируется на моб.уст-ва с помощью JS
      2. **Резиновая верстка**
4. **популярность браузеров** - <https://www.liveinternet.ru/stat/ru/browsers.html>
5. **валидность кода** — это соответствие исходного кода сайта нормам и правилам, описанным Консоциумом Всемирной Паутины или сокращенно W3C. Чтобы проверить свой блог на соответствие этим нормам нужно пройти по ссылке: validator.w3.org. Ввести нужный адрес и посмотреть результаты.

**HTML**

Состоит из Тэгов.

ТЭГИ бывают :

* + 1. Парные <a…..> …….</a>Одниночные(непарны
  1. **Согласно спецификации HTML5** каждый элемент принадлежит к определенной (ноль или более) категории. Каждая из них группирует элементы со схожими характеристиками. Выделяют следующие общие категории:
     1. Мета содержимое
     2. Потоковое содержимое
     3. Секционное содержимое
     4. Заголовочное содержимое
     5. Текстовое содержимое
     6. Встроенное содержимое
     7. Интерактивное содержимоее)

Справочник по HTML **<http://htmlbook.ru/html/img>**

**XHTML**

 ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *e****x****tensible****h****yper****t****ext****m****arkup****l****anguage* — *расширяемый язык гипертекстовой разметки*)

Главное отличие XHTML от HTML заключается в обработке документа. Документы XHTML обрабатываются своим модулем (парсером) аналогично документам XML. В процессе этой обработки ошибки, допущенные разработчиками, не исправляются.

https://ru.wikipedia.org/wiki/XHTML

**мнемоника**

Символ-мнемоника (англ. entity) — это конструкция SGML, которая ссылается на символ из набора символов текстового файла. В HTML предопределено большое количество спецсимволов. Чтобы вставить определённый символ в разметку, нужно вставить определённую ссылку-мнемонику в HTML-структуру.

[**https://ru.wikipedia.org/wiki/Мнемоники\_в\_HTML**](https://ru.wikipedia.org/wiki/Мнемоники_в_HTML)

Ссылки-мнемоники для символов ISO 8859-1

|  |  |  |  |
| --- | --- | --- | --- |
| **Символ** | **Назначение** | **Мнемоника** | **Код** |
|  | [неразрывный пробел](https://ru.wikipedia.org/wiki/%D0%9D%D0%B5%D1%80%D0%B0%D0%B7%D1%80%D1%8B%D0%B2%D0%BD%D1%8B%D0%B9_%D0%BF%D1%80%D0%BE%D0%B1%D0%B5%D0%BB) | &nbsp; | &#160; |
| **¡** | [перевёрнутый восклицательный знак](https://ru.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D0%B5%D0%B2%D1%91%D1%80%D0%BD%D1%83%D1%82%D1%8B%D0%B9_%D0%B2%D0%BE%D1%81%D0%BA%D0%BB%D0%B8%D1%86%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D0%B7%D0%BD%D0%B0%D0%BA) | &iexcl; | &#161; |
| **¢** | [символ цента](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D0%BC%D0%B2%D0%BE%D0%BB_%D1%86%D0%B5%D0%BD%D1%82%D0%B0) | &cent; | &#162; |

**Шрифты**

В работают шрифты wof, wof2

**DOM модель**

DOM (от англ. Document Object Model — «объектная модель документа»)

это не зависящий от платформы и языка программный интерфейс, позволяющий программам и скриптам получить доступ к содержимому HTML-, XHTML- и XML-документов, а также изменять содержимое, структуру и оформление таких документов

https://ru.wikipedia.org/wiki/Document\_Object\_Model

**CSS**

Cascading Style Sheets Каскадные Таблицы Стилей

CSS язык стилей (каскадные таблицы стилей)

Сайт для изучения CSS **http://css.yoksel.ru/svg-bubble/**

**Javascript**

JavaScript (/ˈdʒɑːvɑːˌskrɪpt/; аббр. JS /ˈdʒeɪ.ɛs./) — мультипарадигменный язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили.

**Обьекто-ориентированный клиентский язык**

Является реализацией языка ECMAScript (настоящее название JavaScript)

**XML**

(*e****X****tensible****M****arkup****L****anguage*) — расширяемый [язык разметки](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D1%80%D0%B0%D0%B7%D0%BC%D0%B5%D1%82%D0%BA%D0%B8" \o "Язык разметки).

Рекомендован Консорциумом Всемирной паутины (W3C). Спецификация XML описывает XML-документы и частично описывает поведение XML-процессоров (программ, читающих XML-документы и обеспечивающих доступ к их содержимому). XML разрабатывался как язык с простым формальным синтаксисом, удобный для создания и обработки документов программами и одновременно удобный для чтения и создания документов человеком, с подчёркиванием нацеленности на использование в Интернете. Язык называется расширяемым, поскольку он не фиксирует разметку, используемую в документах: разработчик волен создать разметку в соответствии с потребностями к конкретной области, будучи ограниченным лишь синтаксическими правилами языка.

https://ru.wikipedia.org/wiki/XML

**SGML**

(англ. Standard Generalized Markup Language — стандартный обобщённый язык разметки)

метаязык, на котором можно определять язык разметки для документов

Языки HTML и XML произошли от SGML: HTML до версии 4 включительно был приложением SGML, а XML — это подмножество SGML, разработанное для упрощения процесса машинного разбора документа.

https://ru.wikipedia.org/wiki/SGML

**Регулярные выражения**

**<https://regexone.com>** сайт для тренировки

/^([a-z0-9\_\/-]+)@([a-z0-9\_\/-]+)\.([a-z\.]{2,6})$/; проверка почты

**Консоль открыть**

**Ctrl+Shift+J - хром  
Ctrl+Alt+I - сафари  
F12 - IE9  
Ctrl+Shift+I - FF  
Ctrl+Shift+I - Opera**

**jQuery**

**программный интерфейс приложения, интерфейс прикладного программирования) (англ. application programming interface, API [эй-пи-ай][1] API — это готовый код для упрощения жизни программисту**

<https://jquery.com> оригинальный сайт библиотеки

подключение библиотеки jQuery <script src="js\jquery-3.4.1.min.js"></script>

jquery-docs.ru Рекомендация Гоша Дударь

<http://jquery.page2page.ru/index.php5/Введение_в_jQuery>

<https://html5book.ru/vvedenie-v-jquery/>

<https://www.youtube.com/watch?v=458mWXnWSmA&list=PL0lO_mIqDDFXQk_TTlSO7y7J1l5e-RMgi&index=1>

гоша дударь

**СУБД**

Microsoft SQL Server — система управления реляционными базами данных (РСУБД), разработанная корпорацией Microsoft. Основной используемый язык запросов — Transact-SQL, создан совместно Microsoft и Sybase.

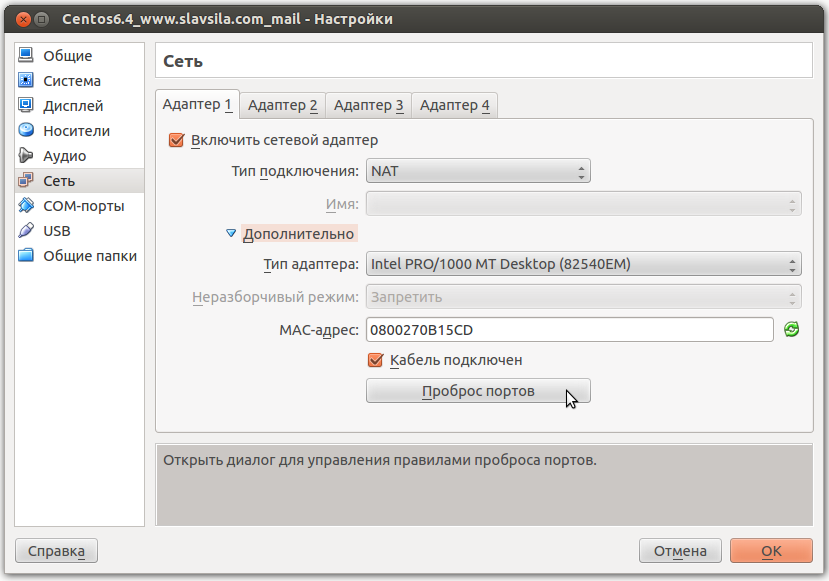
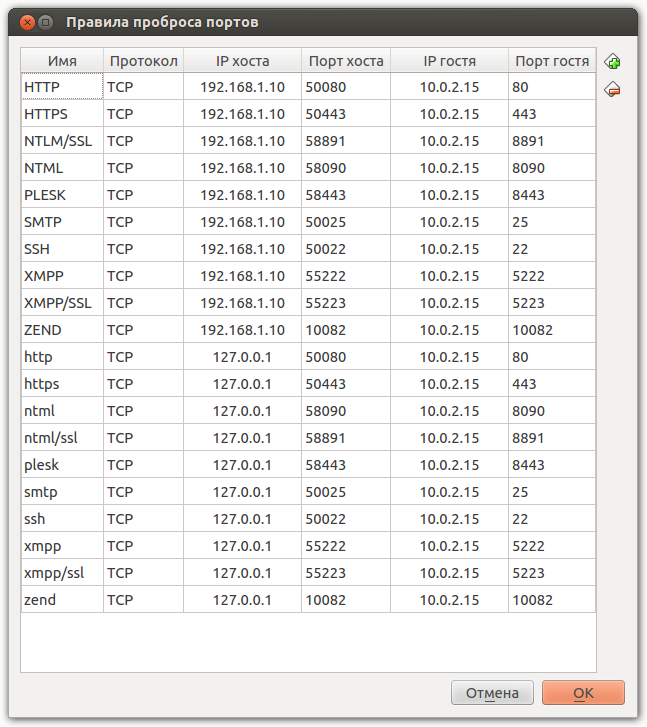
Реализация реляционной СУБД

**https://ru.wikipedia.org/wiki/12\_правил\_Кодда**

**Серверная часть**

**Open Server Panel**

**<https://ospanel.io/download/>**

1. **скачать диструбитив 05.04.2020 ver.5.35.5\_ultimate**
2. **распаковываем в любую папку ( при первом запуске установит) VC++ Microsoft**
3. **ПЕРЕГРУЖАЕМ!!!**
4. **ОШИБКА**
   1. **Файл C:\Windows\system32\drivers\etc\hosts недоступен для записи**
   2. Файл C:\Windows\system32\drivers\etc\hosts недоступен для записи  
      Отключите использование HOSTS файла или настройте права доступа  
      Сбой запуска!
      1. РЕШЕНИЕ: ЗАПУСК OPEN SERVER От имени админситратора
      2. **Файл находится C:\Windows\system32\drivers\etc\hosts**
         1. **attrib.exe -s -r -h -a C:\Windows\system32\drivers\etc\hostsи Запускаем**
         2. **ИЛИ меняем атрибут файла**
5. **OPEN SERVER использует порт:443 для протокола Oracle\VirtualBox\VirtualBox.exe блокирует данный порт**
6. **Поэтому выскакивает ошибка**
7.  

**Или в настройках OPEN SERVER Поменять настройки порта Настройrи-> Сервер-> HTTPS (444 или 50443)**

[**http://integrator.adior.ru/index.php/virtualbox-setup/53-probros-portov-v-virtualnuyu-mashinu-virtualbox**](http://integrator.adior.ru/index.php/virtualbox-setup/53-probros-portov-v-virtualnuyu-mashinu-virtualbox)

**OPEN SERVER -> НАСТРОЙКИ-> СерВер ->**

**(убрать галочку Не вносить изменения в HOST файл)**

**OPEN SERVER -> НАСТРОЙКИ-> Домены -> ничего не вносить**

**Открытие сайтов**

В папке сайта находится файл **.htaccess : его содержимое (последовательность запуска файлов из каталога сайта)**

**<IfModule dir\_module>**

**DirectoryIndex index.html index.php index.htm index.pl**

**</IfModule>**

**17.** **.htaccess**

Файл .htaccess (англ. hypertext access) используется для простой и удобной настройки веб-сервера на котором хранится сайт пользователя. Соответственно меняя настройку веб-сервера, мы сможем поменять работу сайта. Как правило, файл .htaccess находится в корневом каталоге, а его действие распространяется на весь сайт и на все подкаталоги. Если же в другом каталоге содержится свой .htaccess, то он будет действовать только на свой каталог и подкаталоги.

**SQL**

**structured query language** — «язык структурированных запросов»

[декларативный](https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D0%BA%D0%BB%D0%B0%D1%80%D0%B0%D1%82%D0%B8%D0%B2%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" \o "Декларативное программирование) [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), применяемый для создания, модификации и управления данными в [реляционной базе данных](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D1%8B%D0%B5_%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), управляемой соответствующей [системой управления 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).

<https://ru.wikipedia.org/wiki/SQL>

**SQL - язык запросов для управления СУБД.**

**MySQL - это одна из множества других СУБД.**

https://myrusakov.ru/sql-mysql-raznica.html

**MySQL**

Разработку и поддержку MySQL осуществляет корпорация Oracle <http://www.mysql.ru/>

запуск из панели задач – phpMyAdmin имя root пароль пустой

**Предикаторы**

Выражение в запросе MySQL после WHERE примеры BETWEEN 1 AND 5; NOT BETWEEN 1 AND 5;id IN (5,7);NOT IN (5,7)

**ПОДКЛЮЧЕНИЯ BEGET.RU**

**beget.ru логин: jjohn057 пароль: alfa3500**

Для этого необходимо перейти в раздел MySQL, нажать на плюс справа от выбранной базы добавить IP-адрес и пароль с которых необходимо разрешить подключение.

**Сервер для подключения сайтов:** localhost (**сайт расположен на beget.ru**)

**$mysqli = @new mysqli(‘localhost’, ‘имя\_пользователя\_совпадает\_имя\_базы\_данных\_на+\_сервере\_ beget.ru’,’пароль\_базы\_данных’,’ имя\_базы\_данных\_на+\_сервере\_ beget.ru’);** // создаем обьект(экземпляр) встроенного класса MYSQLI с параметрами и подключаемся к базе

**Сервер для внешних подключений:** jjohn057.beget.tech (**сайт расположен в инете)**

**Имя пользователя:**  
Совпадает с именем БД

($servername = "jjohn057.beget.tech"; $username = "имя\_базы данных"; $password = "пароль базы"; $dbname= "имя\_базы";)

Для подключения к **phpMyAdmin: <https://bane.beget.com/phpMyAdmin>**

База данных перемещается из отладки через ИМПОРТ-ЭКСПОРТ

Имена баз данных mySQL:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Имя базы | Пароль | Пароль внешний | Доступ с BEGET.RU | Доступ из инета | Сайт прилинкованный |  |
| jjohn057\_din\_mir | 0@}MH}A2-xnnu7 | M89H&xsl | localhost | jjohn057.beget.tech | www.dinamik-mir.ru | dinamiki-mir.ru\public\_html\config |
| jjohn057\_din\_ua | %7vu4S{0-nwy3 | g%p6kRV9 | localhost | jjohn057.beget.tech |  |  |
| jjohn057\_ilina | 19860328 | 19860328 | localhost | jjohn057.beget.tech | Ильина.рус |  |
| jjohn057\_kirpich | qHqUxh14-7h2^h | %AUr6RaN | localhost | jjohn057.beget.tech |  | config\_bd.php |
| jjohn057\_proba | 19860328 | 19860328 | localhost | jjohn057.beget.tech |  |  |

# Visual Studio

# Устанавливаем Visual Studio 2019

# <https://visualstudio.microsoft.com/ru/downloads/?rr=https%3A%2F%2Fdocs.microsoft.com%2Fru-ru%2Fvisualstudio%2Fget-started%2Fvisual-basic%2Ftutorial-console%3Fview%3Dvs-2019>

## Обзор Visual Studio 2019

**Visual Studio** – это интегрированная среда разработки (IDE) от компании Microsoft.

С помощью Visual Studio можно разрабатывать:

* Классические приложения для компьютера под управлением операционной системы Windows;
* Мобильные приложения (Windows, iOS, Android);
* Web-приложения;
* Облачные приложения;
* Различные расширения для Office, SharePoint, а также создание собственных расширений для Visual Studio;
* Игры;
* Базы данных SQL Server и SQL Azure.

В Visual Studio Вы можете использовать следующие технологии и языки программирования: .NET, Node.js, C, C#, C++, Python, Visual Basic, F#, JavaScript.

**Visual Studio (VSCode)**

https://issue.life/questions/30527522

* **NET Core (Mac / Linux / Windows)**
* **NET Framework (только для Windows)** программная платформа, выпущенная компанией Microsoft в 2002 году. Основой платформы является общеязыковая среда исполнения Common Language Runtime (CLR), которая подходит для разных языков программирования. Функциональные возможности CLR доступны в любых языках программирования, использующих эту среду.

[**https://ru.wikipedia.org/wiki/.NET\_Framework**](https://ru.wikipedia.org/wiki/.NET_Framework)

В настоящее время .NET Framework получает развитие в виде .NET Core, изначально предполагающей кроссплатформенную разработку и эксплуатацию.

Состоят из исполняющей среды Common Language Runtime (CLR) для .NET Framework

Core Language Runtime

Объектные классы .NET, доступные для всех поддерживаемых языков программирования, содержатся в библиотеке Framework Class Library (FCL). В FCL входят классы Windows Forms, ADO.NET, ASP.NET, Language Integrated Query, Windows Presentation Foundation, Windows Communication Foundation и другие. Ядро FCL называется Base Class Library (BCL).

**редактор кода для Linux, OS X и Windows**

# Visual Studio (VSCode) - это кросс-платформенный (Linux, Mac OS, Windows) редактор, который может быть дополнен плагинами в соответствии с вашими потребностями.

**Visual Studio Code - это что-то вроде Notepad++.** При чём в этом редакторе **нет поддержки C++**

**Microsoft Expression Blend**

**Microsoft Expression Blend — программа от корпорации Microsoft, предназначенная для разработки дизайна веб-интерфейсов и графических настольных приложений, совмещает в себе особенности этих двух типов приложений. Интерактивный WYSIWYG-редактор для дизайна интерфейсов, основанных на XAML-приложениях, для Windows Presentation Foundation и приложения Silverlight. Входит в состав пакета графических программ Microsoft Expression Studio. Expression Blend поддерживает WPF, в котором используется метод OpenType и ClearType, основанный на 2D- и 3D-элементах управления с аппаратным ускорением через DirectX.**

**Импортируется в проект из Photoshop и Illuctration со всеми слоями и функциями**

**C# C++**

**Отличии очень большие**

**C# (Си шарп)**— объектно-ориентированный язык программирования с автоматическим управлением памятью, предназначенный для программирования на платформе .NET.

[**https://ru.wikipedia.org/wiki/C\_Sharp**](https://ru.wikipedia.org/wiki/C_Sharp)

**C++** — кросс-платформенный язык программирования общего назначения, совместимый в большинстве случаев с C.

В С++ существует понятие пространство имен. Это пространство определяет некую область, на которую приходятся действия оператора или функции. Для того что бы использовать оператор, находящийся в определенном пространстве,необходимо подключить это пространство в свою программу. Например подключить просторанства std ( using namespace std;

**GNU**

*GNU’s Not UNIX* — «GNU — не [Unix](https://ru.wikipedia.org/wiki/Unix)»[[1]](https://ru.wikipedia.org/wiki/GNU#cite_note-:0-1)) — [свободная](https://ru.wikipedia.org/wiki/%D0%A1%D0%B2%D0%BE%D0%B1%D0%BE%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)[[2]](https://ru.wikipedia.org/wiki/GNU#cite_note-2) [Unix-подобная операционная система](https://ru.wikipedia.org/wiki/Unix-%D0%BF%D0%BE%D0%B4%D0%BE%D0%B1%D0%BD%D0%B0%D1%8F_%D0%BE%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), разрабатываемая [Проектом GNU](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B5%D0%BA%D1%82_GNU)

Проект GNU (англ. The GNU Project) — проект по разработке свободного программного обеспечения (СПО), является результатом сотрудничества множества отдельных проектов. Проект был запущен известным программистом и сторонником СПО Ричардом Столлманом 27 сентября 1983 года в Массачусетском технологическом институте.[1] Изначальной целью проекта было «разработать достаточно свободного программного обеспечения <…>, чтобы можно было обойтись без программного обеспечения, которое не является свободным».[2] Чтобы этого достичь, проект в 1984 году приступил к разработке операционной системы GNU (GNU является рекурсивным акронимом, расшифровывающимся, как «GNU is Not Unix»). Эта цель была достигнута в 1992 году, когда последний пробел в ОС GNU — ядро системы — был заполнен сторонней разработкой, ядром Linux, которое было выпущено как свободное программное обеспечение в соответствии с лицензией GNU GPL v2.

**Основатель Ричард Столлман**

Текущая работа проекта GNU включает в себя разработку программного обеспечения, повышение осведомлённости, проведение политических кампаний и раздачу новых материалов.

https://ru.wikipedia.org/wiki/Проект\_GNU

**Linux**

**Установка**

16.02.2020 Ubuntu180404LST Name Jjohn

**Общие описание**

семейство [Unix-подобных операционных систем](https://ru.wikipedia.org/wiki/Unix-%D0%BF%D0%BE%D0%B4%D0%BE%D0%B1%D0%BD%D0%B0%D1%8F_%D0%BE%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) на базе [ядра Linux](https://ru.wikipedia.org/wiki/%D0%AF%D0%B4%D1%80%D0%BE_Linux), включающих тот или иной набор утилит и программ проекта [GNU](https://ru.wikipedia.org/wiki/GNU), и, возможно, другие компоненты. Как и ядро Linux, системы на его основе как правило создаются и распространяются в соответствии с моделью разработки [свободного и открытого программного обеспечения](https://ru.wikipedia.org/wiki/FOSS). Linux-системы распространяются в основном бесплатно в виде различных [дистрибутивов](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D1%81%D1%82%D1%80%D0%B8%D0%B1%D1%83%D1%82%D0%B8%D0%B2_Linux) — в форме, готовой для установки и удобной для сопровождения и обновлений, — и имеющих свой набор [системных](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%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) и [прикладных](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) компонентов, как свободных, так, возможно, и собственнических.

Официальным [логотипом](https://ru.wikipedia.org/wiki/%D0%9B%D0%BE%D0%B3%D0%BE%D1%82%D0%B8%D0%BF) и [талисманом](https://ru.wikipedia.org/wiki/%D0%A2%D0%B0%D0%BB%D0%B8%D1%81%D0%BC%D0%B0%D0%BD) Linux является [пингвин](https://ru.wikipedia.org/wiki/%D0%9F%D0%B8%D0%BD%D0%B3%D0%B2%D0%B8%D0%BD) [Tux](https://ru.wikipedia.org/wiki/Tux), созданный в 1996 году [Ларри Юингом](https://ru.wikipedia.org/wiki/%D0%AE%D0%B8%D0%BD%D0%B3,_%D0%9B%D0%B0%D1%80%D1%80%D0%B8)[[18]](https://ru.wikipedia.org/wiki/Linux#cite_note-18). Торговая марка «Linux» принадлежит создателю и основному разработчику ядра [Линусу Торвальдсу](https://ru.wikipedia.org/wiki/%D0%A2%D0%BE%D1%80%D0%B2%D0%B0%D0%BB%D1%8C%D0%B4%D1%81,_%D0%9B%D0%B8%D0%BD%D1%83%D1%81).

**<https://ru.wikipedia.org/wiki/Linux>**

Большинство пользователей для установки Linux используют дистрибутивы, включающие не только набор программ, но и решающие ряд задач по обслуживанию, объединённых едиными системами установки, управления и обновления пакетов, настройки и поддержки.

**Самые распространённые в мире дистрибутивы (2017)[61]: Linux Mint, Ubuntu, Debian, Mageia, Fedora, OpenSUSE, ArchLinux, CentOS, PCLinuxOS, Slackware, Gentoo**.

2**Большинство серверов (хостинги, хранилища и прочее) используют в качестве основной ОС Линукс.** Это связано с тем, что она бесплатна, проще настраивается и более стабильна. Однако среди обычных пользователей бесспорное лидерство занимает Windows. Лишь 5% населения пользуются Линуксом в качестве домашней операционной системы.

3 **Linux является многозадачной и многопользовательской операционной системой**. Windows в этом плане также сильно преуспел, но все же его возможности не очень удобные для работы нескольких пользователей

4 В Linux нету жестких дисков, как на Windows. Здесь все хранится в едином каталоге, который начинается с «/» (корневой каталог, root directory). Файлы пользователей лежат в каталогах «/home/name\_users/».

**5 Программные оболочки bash или tcsh**. С помощью них Linux выполняет команды пользователей. Отдаваемая команда относится к текущему каталогу (current directory). Домашний каталог можно обозначать символом «~» (тильда). Это аналог «/home/name\_users/»

**6 Файловые системы в Linux отличны от Windows:**

* ext4 — журналируемая файловая система (стандарт для Linux)
* btrfs — новая файловая система, основанная на структурах B-деревьев
* xfs — данную файловую систему отличает достаточно высокая производительность в части записи и чтения данных.

Стоит отметить, что Линукс поддерживает очень много файловых систем (в том числе Windows-ие NTFS, FAT32). Напомню, что Windows не поддерживает ext4. Про файловые системы Linux будет рассказано более подробно в отдельной статье.

7 **Существует два вида дистрибутивов Linux,** которые делятся по способу организации и управления программным обеспечением (пакетами)

1. **DEB — используются в дистрибутиве Debian** и во всех основанных на нем дистрибутивах
2. **RPM — менеджер пакетов, использующийся в дистрибутиве Red Hat**, а также во многих других популярных дистрибутивах (Fedora)

8 **Процессы разделены по ресурсам.** Чтобы добавить новое устройств нужен драйвер. Его можно написать самостоятельно и запустить в ядре. Драйверы устройств либо интегрируются непосредственно в ядро, либо добавляются в виде модулей, загружаемых во время работы системы.

9 **В Linux все считается файлами:** жесткие диски, разделы, папки, приводы оптических дисков, съемные носители информации.

5. **Интерфейсы Linux**

Поскольку дистрибутивов много, то и различных версий оформления рабочего стола/панели инструментов также много. Однако в Линуксе придумано несколько стандартов, которые поддерживаются популярными версиями ОС.

* **GNOME** (GNU Network Object Model Environment) и **Unity**. Концепцию значков лаунчеров и отображения в стиле панели мониторинга для выбора
* **KDE** (K Desktop Environment). Традиционная среда рабочего стола, но она имеет огромное количество функций и основной набор приложений
* **Cinnamon**. Основан на GNOME. Традиционная среда рабочего стола, которая похожа на Windows 7.
* **LXDE** (Lightweight X11 Desktop Environment). Легкая и приятная среда.
* **MATE** является продолжением классического интерфейса GNOME
* **XFCE** быстрая и легкая

**Linux Mint**

Новым пользователям, мигрировавшим с [Windows](https://lifehacker.ru/live-linux-distribution/), однозначно стоит установить Linux Mint. Некогда он был популярнейшим дистрибутивом Linux, но сейчас его перегнали MX Linux и Manjaro. Тем не менее Mint выгодно отличается от них своей стабильностью и надёжностью. Это очень простая в работе система, основанная на Ubuntu.

Linux Mint оснащён лёгким и понятным интерфейсом (оболочка Cinnamon для современных компьютеров и MATE для старых машин) и удобным менеджером приложений, так что у вас не возникнет проблем с поиском и установкой программ.

**openSUSE**

стабильный и мощный дистрибутив

**Manjaro**

для тех, кто хочет иметь свежайшее ПО

**Tails**

На базе Debian выпущена еще одна система – Tails. Она предназначена для тех, кому важно сохранять анонимность в сети. Из достоинств этой системы: для нее не нужно сильное оборудованние, она стабильна. Tails идеально подходит для серфинга в Интернете и решения повседневных задач.

**Debian**

Эта версия любима за счет своей стабильности и безопасности. Команда разработчиков впечатляет, но обновления для системы выпускаются редко. Эта система удобна для удаленного администрирования, поэтому ее часто ставят на серверы. Для новичков сборка сложна, в ней слишком много настроек и программ. При этом дистрибутив имеет несколько положительных качеств: стабильность; поддерживается множество архитектур; безопасность при использовании; более тысяч пакетов программ; легко обновляется; быстро и эффективно работает даже на устаревших компьютерах. В отличие от других версий дистрибутивов, сразу после установки Debian придется настроить. Разобраться в количестве опций сможет только опытный пользователь. Популярность дистрибутива сохраняется только за счет его стабильности, но с точки зрения интерфейса и удобства пользования он устарел. Обновления операционной системы выходят редко, но их загрузка и установка крайне просты.

**ArchLinux**

Современный и быстро обновляющейся дистрибутив.

**Linux Ubuntu**

Этот дистрибутив является самым распространенным и популярным. Практически все пользователи Linux хотя бы раз попробовали его. Система идеально подходит для новичков, которые только хотят ознакомиться с возможностями дистрибутивов. В интерфейсе нет ничего лишнего, в том числе разработчики избавились и от терминала. Для неопытных пользователей может стать недостатком использование командной строки для работы с системой. Преимущества Ubuntu: бесплатное распространение, программы и компоненты можно также свободно загружать; процесс установки не отнимает более 10 минут; в интерфейсе легко разобраться, он понятен; без разрешения пользователя в системе ничего не происходит, поэтому риск заражения вирусами минимален; можно использовать на одном компьютере с Windows, в систему вшита возможность мультизагрузки; в сборку входит достаточный набор программ; сообщества и форумы позволяют решить любую пролему. Основной недостаток этой версии – нестабильность работы. От дистрибутива многие отказываются из-за сбоев, которые сопровождают практически каждое обновление системы. Новые версии часто выдают ошибки, с которыми другие пользователи ранее не сталкивались. После этой версии Linux будет сложно использовать другие дистрибутивы.

https://ubuntu.ru/get

**Ubuntu 18.04 LTS (Bionic Beaver) LTS (Long Term Support) - это словосочетание означает версию системы с длительной поддержкой, для которой обновления будут выходить в течении 5 лет (до апреля 2023г.).**

**Дата выхода: 26.04.2018г.; четвертый исправленный выпуск: 12.02.2020г.**

**Ubuntu Desktop - операционная система для настольных компьютеров и ноутбуков.**

**Ubuntu Server - операционная система для серверов.**

LTS (Long Term Support) - это словосочетание означает версию системы с длительной поддержкой, для которой обновления будут выходить в течении 5 лет (до апреля 2023г.).

Ubuntu Advantage - платная техподдержка от партнеров Canonical, в том числе обновление системных библиотек в течении 10 лет (до апреля 2028г.).

**Fedora**

Если вы хотите одним из первых протестировать новинки из мира Linux, то вам стоит установить дистрибутив Fedora. Он также связан с Red Hat, которая использует эту версию в качестве бесплатного полигона для обкатки новинок. Сам основатель Linux Торвальдс Линус использует этот дистрибутив в качестве основного, а его выбору не доверять не стоит. Единственным недостатком этой системы является лишь небольшое число ошибок после обновления. Дистрибутив хорош для использования дома. Он не выходит из строя после внедрения новинок. Он идет в ногу со временем и подходит для компьютеров с разной мощностью.

**Репозиторий Linux**

В Linux все программы (пакеты) хранятся в репозиториях, который представляет из себя сервер, где программисты выкладывают софт. Здесь можно скачать самые последние и актуальные версии программ.

Для Линукса репозиторий это аналог Play Market для Android.

**FreeBSD**

FreeBSD - это продвинутая операционная система для x86-совместимых, DEC Alpha, IA-64, PC-98 и UltraSPARC архитектур. Она происходит от BSD, версии UNIX®, разработанной в Университете Калифорнии, Беркли. Разработку и поддержку осуществляет большая группа людей. Версии для прочих платформ находятся на разных стадиях разработки.

http://www.freebsd.org.ru

**https://ru.wikipedia.org/wiki/FreeBSD**

**НАТИВНЫЕ И КРОССПЛАТФОРМЕННЫЕ ПРИЛОЖЕНИЯ**

**Нативные приложения (от англ. native - родной)** разрабатываются под конкретную аппаратно-программную платформу и пишутся на языках, созданных для данной платформы. И iOs, и Android имеют свои SDK (от англ. software development kit — набор средств разработки) и свой стек технологий, завязанные на определенный язык программирования. Например, родными языками для Android являются Java и Kotlin, для iOS, соответственно - Swift и Objective-C.

Нативные приложения создаются специально для запуска на целевой платформе - с поддержкой всех нативных технологий и аппаратных возможностей конкретной платформы.

Как следует из названия, **кроссплатформенность подразумевает создание приложений, которые могут работать в различных операционных системах**. После написания кода приложения его можно развернуть на разных устройствах и платформах, не беспокоясь о проблемах несовместимости. Это универсальный подход, который широко используется для экономии времени и денег на разработку. Часто для этого используются специализированные кроссплатформенные фреймворки.

**PHP**

PHP (/pi:.eɪtʃ.pi:/ англ. PHP: Hypertext Preprocessor — «PHP: препроцессор гипертекста»; первоначально Personal Home Page Tools[6] — «Инструменты для создания персональных веб-страниц») — скриптовый язык[7] общего назначения, интенсивно применяемый для разработки веб-приложений

Web Developer Blog

**PDO PHP (PHP Data Objects)**

Расширение Объекты данных PHP (PDO)

**C:\OSPanel\modules\php\PHP-7.2-x64 php.ini**

PDO – PHP Data Objects – это библиотека, которая содержит набор методов, по работе с различными базами данных, при этом доступ к конкретной базе данных, осуществляется при помощи соответствующего драйвера, который обязательно должен быть подключен в интерпретаторе языка PHP, Таким образом PDO, обеспечивает абстракцию доступа к базам данным, а это значит, что независимо от того, какая база используется, для работы с данными, Вы можете использовать одни и те же методы. Что очень удобно для реализации поддержки в создаваемом скрипте различных баз данных.

PDO предлагает только объектно-ориентированных подход и внедрен в язык PHP, начиная с версии 5.1

**https://webformyself.com/vvedenie-v-pdo/**

[**http://php.net/manual/ru/class.pdo.php**](http://php.net/manual/ru/class.pdo.php)

**GIT**

**Система контроля версий**

**<https://githowto.com/ru>**

**http://kb.etersoft.ru/Работа\_с\_git изучение (РУССКОЕ ОПИСАНИЕ СУПЕР!)**

[**http://mellarius.ru/git**](http://mellarius.ru/git) **изучение (РУССКОЕ ОПИСАНИЕ 4)**

**Git** (произнoсится «гит»[[7]](https://ru.wikipedia.org/wiki/Git#cite_note-7)) — распределённая [система управления версиями](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%B2%D0%B5%D1%80%D1%81%D0%B8%D1%8F%D0%BC%D0%B8). <https://ru.wikipedia.org/wiki/Git>

[**https://tproger.ru/translations/beginner-git-cheatsheet/**](https://tproger.ru/translations/beginner-git-cheatsheet/)

**<https://git-scm.com/download/win>**

**UNTRAKED – файлы не под версионным контролем**

**Глобальные настройки хранятся в файле .gitconfig в каталоге пользвателя C:\Users\John**

**Comit – значит зафиксировать изменения**

git config --global user.name "John"

git config --global user.email “jjohn057@gmail.com”

git config –list (выводит глобальные настройки)

core.symlinks=false

core.autocrlf=true

core.fscache=true

color.diff=auto

color.status=auto

color.branch=auto

color.interactive=true

help.format=html

rebase.autosquash=true

http.sslbackend=openssl

http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt

credential.helper=manager

core.editor="C:\\Program Files (x86)\\Notepad++\\notepad++.exe" -multiInst -notabbar -nosession -noPlugin

**user.name=John**

**user.email=jjohn057@gmail.com**

**Config file location**

--global use global config file

--system use system config file

--local use repository config file

--worktree use per-worktree config file

-f, --file <file> use given config file

--blob <blob-id> read config from given blob object

Action

--get get value: name [value-regex]

--get-all get all values: key [value-regex]

--get-regexp get values for regexp: name-regex [value-regex]

--get-urlmatch get value specific for the URL: section[.var] URL

--replace-all replace all matching variables: name value [value\_regex]

--add add a new variable: name value

--unset remove a variable: name [value-regex]

--unset-all remove all matches: name [value-regex]

--rename-section rename section: old-name new-name

--remove-section remove a section: name

-l, --list list all

-e, --edit open an editor

--get-color find the color configured: slot [default]

--get-colorbool find the color setting: slot [stdout-is-tty]

Type

-t, --type <> value is given this type

--bool value is "true" or "false"

--int value is decimal number

--bool-or-int value is --bool or --int

--path value is a path (file or directory name)

--expiry-date value is an expiry date

Other

-z, --null terminate values with NUL byte

--name-only show variable names only

--includes respect include directives on lookup

--show-origin show origin of config (file, standard input, blob, command line)

--default <value> with --get, use default value when missing entry

**Vi или Vim редактор**

https://help.ubuntu.ru/wiki/vim

**<https://xakep.ru/2017/10/03/vim/>**

Vim — свободный текстовый редактор, созданный на основе более старого vi. Ныне это один из мощнейших текстовых редакторов с полной свободой настройки и автоматизации, возможными благодаря расширениям и надстройкам. Пользовательский интерфейс Vim’а может работать в чистом текстовом режиме. [Википедия](https://ru.wikipedia.org/wiki/Vim)

**GitHub**

**Слоган сервиса — «Social Coding» — на русский можно перевести как «Пишем код вместе»**

[**https://github.com**](https://github.com) **Login:John057 password: Alfa3500 e-mail:** [**jjohn057@gmail.com**](mailto:jjohn057@gmail.com)

Organization name: John3500 Billing email: [jjohn057@gmail.com](mailto:john057@gmail.com)

<https://github.com/account/unverified-email>

Можно работать через переводчик Google

**Git** (произнoсится «гит»[[7]](https://ru.wikipedia.org/wiki/Git#cite_note-7)) — распределённая [система управления версиями](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%B2%D0%B5%D1%80%D1%81%D0%B8%D1%8F%D0%BC%D0%B8).

[**https://ru.wikipedia.org/wiki/Git**](https://ru.wikipedia.org/wiki/Git)

Среди проектов, использующих Git — [ядро Linux](https://ru.wikipedia.org/wiki/%D0%AF%D0%B4%D1%80%D0%BE_Linux), [Swift](https://ru.wikipedia.org/wiki/Swift_(%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)), [Android](https://ru.wikipedia.org/wiki/Android), [Drupal](https://ru.wikipedia.org/wiki/Drupal), [Cairo](https://ru.wikipedia.org/wiki/Cairo), [GNU Core Utilities](https://ru.wikipedia.org/wiki/GNU_Coreutils), [Mesa](https://ru.wikipedia.org/wiki/Mesa_3D), [Wine](https://ru.wikipedia.org/wiki/Wine), [Chromium](https://ru.wikipedia.org/wiki/Chromium), [Compiz Fusion](https://ru.wikipedia.org/wiki/Compiz_Fusion), [FlightGear](https://ru.wikipedia.org/wiki/FlightGear), [jQuery](https://ru.wikipedia.org/wiki/JQuery), [PHP](https://ru.wikipedia.org/wiki/PHP), [NASM](https://ru.wikipedia.org/wiki/NASM), [MediaWiki](https://ru.wikipedia.org/wiki/MediaWiki), [DokuWiki](https://ru.wikipedia.org/wiki/DokuWiki), [Qt](https://ru.wikipedia.org/wiki/Qt), ряд дистрибутивов [Linux](https://ru.wikipedia.org/wiki/Linux).

Программа является свободной и выпущена под лицензией [GNU GPL](https://ru.wikipedia.org/wiki/GNU_General_Public_License) версии 2. По умолчанию используется TCP порт 9418.

Терминальная версия Windows64

Name: John057 e-mail: [48690689+John057@users.noreply.github.com](mailto:48690689+John057@users.noreply.github.com)

48690689+John057@users.noreply.github.com

Aton – Редактор GitHub

**Composer(getcomposer.org)**

это относительно новый и уже достаточно популярный менеджер зависимостей для PHP. Вы можете описать от каких библиотек зависит ваш проект и Composer установит нужные библиотеки за вас! Причём Composer — это не менеджер пакетов в классическом понимании. Да, он оперирует с сущностями, которые мы будем называть «пакетами» или библиотеками, но устанавливаются они внутрь каждого проекта отдельно, а не глобально (это одно из основных отличий от старого-доброго PEAR).

**Composer** — это [пакетный менеджер](https://ru.wikipedia.org/wiki/%D0%9F%D0%B0%D0%BA%D0%B5%D1%82%D0%BD%D1%8B%D0%B9_%D0%BC%D0%B5%D0%BD%D0%B5%D0%B4%D0%B6%D0%B5%D1%80) уровня приложений для языка программирования [PHP](https://ru.wikipedia.org/wiki/PHP), который предоставляет средства по управлению зависимостями в PHP-приложении. Composer разработали и продолжают поддерживать два программиста Nils Adermann и Jordi Boggiano.

Оригинал English <https://getcomposer.org/download/>

Русская версия

**http://getcomposer.ucoz.org/publ/kniga/skhema\_composer\_json/2-1-0-5**

**http://getcomposer.ucoz.org/publ/kniga/nachalo\_raboty/2-1-0-1**

**Исключение**

Исключение — это объект встроенного в PHP класса Exception ([мануал по Exception](http://php.net/manual/ru/class.exception.php)) или его наследника. Объект исключения содержит подробности о причинах ошибки. Также, в PHP есть еще другие классы исключений, которые ты можешь использовать: <http://php.net/manual/ru/spl.exceptions.php>

В PHP7 определение расширено: исключения не обязаны наследоваться от Exception, это может быть любой класс, реализующий интерфейс [Throwable](http://php.net/manual/en/class.throwable.php), а так как сам Exception его реализует, то его наследники по-прежнему будут работать

**ООП (обьектно-ориентированное программирование)**

ООП это не конструкции языка, это стиль программирования.

**Объе́ктно**-**ориенти́рованное программи́рование** (**ООП**) — методология **программирования**, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определённого класса, а классы образуют иерархию наследования.

1. Классы
   1. констукторы деструкторы методы
      1. Дестру́ктор — специальный метод класса, служащий для деинициализации объекта (например освобождения памяти).
         1. **Деструктор класса PHP** ООП — это специальная функция (принадлежащая к группе методов, известных как магические методы), которая автоматически вызывается при удалении объекта. ... **Деструктор** выполняет обратную функцию, во время удаления объекта он освобождает память и ресурсы сервера.

**Паттерны**

Шаблон проектирования или **паттерн (англ. design pattern)** в разработке программного обеспечения — повторяемая архитектурная конструкция, представляющая собой решение проблемы проектирования в рамках некоторого часто возникающего контекста.

Ссылка на книги J:\Torrent\Программирование Основы\Паттерны

Основная и главная **книга банды четырех «паттерны проектирования»**

**Фреймворк**

Фреймворк» отличается от понятия библиотеки тем, что библиотека может быть использована в программном продукте просто как набор подпрограмм близкой функциональности, не влияя на архитектуру программного продукта и не накладывая на неё никаких ограничений. «Фреймворк» же диктует правила построения архитектуры приложения, задавая на начальном этапе разработки поведение по умолчанию — «каркас», который нужно будет расширять и изменять согласно указанным требованиям.

**<https://saikt-online.ru/chto-takoe-frejmvorki-i-dlya-chego-oni-nuzhny/>**

1. **фреймворки (framework) для PHP:** 
   1. **Yii 2.0 (лучший по мнению Михаила Русакова) <https://www.yiiframework.com/doc/guide/2.0/ru>**
   2. **Laravel** [**https://laravel.ru**](https://laravel.ru)
   3. CakePHP

**фрэймворк Bootstrap**

**Bootstrap** (также известен как Twitter Bootstrap[1][2][3]) — свободный набор инструментов для создания сайтов и веб-приложений. Включает в себя HTML- и CSS-шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая JavaScript-расширения.

Bootstrap использует современные наработки в области CSS и HTML, поэтому необходимо быть внимательным при поддержке старых браузеров

<https://ru.wikipedia.org/wiki/Bootstrap_(фреймворк)>

www.getbootstrap.com

**Сущность**

Сущность - это реальный или представляемый объект, информация о котором должна сохраняться и быть доступна.

**Объе́кт** в [программировании](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) — некоторая сущность в цифровом пространстве, обладающая определённым состоянием и поведением, имеющая определенные свойства ([атрибуты](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%BB%D0%B5_%D0%BA%D0%BB%D0%B0%D1%81%D1%81%D0%B0)) и операции над ними ([методы](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_(%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)))[[1]](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)#cite_note-def-1).

<https://ru.wikipedia.org/wiki/Объект_(программирование)>

Источник этого непонимания лежит в самом начале науки - у Аристотеля. Аристотель выделял два вида сущности: первую сущность и вторую сущность. **Первая сущность** - это что-то конкретное, что существует в единственном экземпляре (конкретный Иван Иванович Иванов - это первая сущность). **Вторая сущность** - это вид или род, к которому относится первая сущность. Иван Иванов (первая сущность) - программист (относится к виду программистов из рода человекоподобных). Сам **"Программист"** в данном случае - это вторая сущность по Аристотелю.Когда программисты используют слово "сущность", они, как правило, имеют в виду "первую сущность" по Аристотелю, то есть нечто уникальное, что нельзя ни с какой другой первой сущностью сравнить.

**Наследование**

Наследование (англ. inheritance) — концепция объектно-ориентированного программирования, согласно которой абстрактный тип данных может наследовать данные и функциональность некоторого существующего типа, способствуя повторному использованию компонентов программного обеспечения.

https://ru.wikipedia.org/wiki/Наследование\_(программирование)

**Полиморфизм**

https://ru.wikipedia.org/wiki/Полиморфизм\_(информатика)

в языках программирования и теории типов — способность функции обрабатывать данные разных типов[

**Инкапсуляция**

https://ru.wikipedia.org/wiki/Инкапсуляция\_(программирование)

в информатике упаковка данных и функций в единый компонент.

**В общем случае в разных языках программирования термин «инкапсуляция» относится к одной или обеим одновременно следующим нотациям:**

**механизм языка, позволяющий ограничить доступ одних компонентов программы к другим;**

**языковая конструкция, позволяющая связать данные с методами, предназначенными для обработки этих данных.**

**PSR (PHP standart recommendation) стандартов**

Всего на момент написания статьи существует 17 стандартов, причем 9 из них являются утвержденными, 8 находятся в стадии проекта, активно обсуждаются, 1 стандарт не рекомендован к использованию.

Теперь перейдем непосредственно к описанию каждого стандарта. Заметьте, что я не буду здесь подробно разбирать каждый стандарт, скорее это небольшое введение. Также в статье будут рассматриваться только те стандарты PSR, которые официально приняты, т.е. находятся в статусе Accepted.

PSR-1. Основной стандарт кодирования

Он представляет собой наиболее общие правила, такие как, например, использование тегов PHP, кодировка файлов, разделения места объявления функции, класса и места их использования, именование классов, методов.

PSR-2. Руководство по стилю кода

Является продолжением первого стандарта и регулирует вопросы использования в коде табуляции, переводов строк, максимальную длину строк кода, правила оформления управляющих конструкций и т.д.

PSR-3. Интерфейс протоколирования.

Этот стандарт разработан для того, чтобы обеспечить (журналирование) логирование в приложениях, написанных на PHP.

PSR-4. Стандарт автозагрузки

Это, наверное, самый важный и нужный стандарт, которому будет посвящена отдельная, подробная статья. Классы, которые реализуют PSR-4, могут быть загружены единым автозагрузчиком, что позволяет частям и компонентам из одного фреймворка или библиотеки быть использованными в других проектах.

PSR-6. Интерфейс кеширования

Кэширование используется для повышения производительности системы. И PSR-6 позволяет стандартно сохранять и извлекать данные из кэша, используя унифицированный интерфейс.

PSR-7. Интерфейс HTTP-сообщений

При написании мало-мальски сложных сайтов на PHP, почти всегда приходиться работать с HTTP заголовками. Конечно, язык PHP предоставляет нам уже готовые возможности для работы с ними, такие как суперглобальный массив $\_SERVER, функции header(), setcookie() и т.д., однако их ручной разбор чреват ошибками, да и не всегда можно учесть все нюансы работы с ними. И вот, чтобы облегчить работу разработчику, а также сделать единообразным интерфейс взаимодействия с HTTP протоколом был принят данный стандарт. Более подробно об этом стандарте я расскажу в одной из следующих статей.

PSR-11. Интерфейс контейнера

При написании PHP программы часто приходится использовать сторонние компоненты. И чтобы не заблудиться в этом лесу зависимостей были придуманы различные методы управления зависимостями кода, зачастую несовместимые между собой, которые данный стандарт и приводит к общему знаменателю.

PSR-13. Гипермедиа ссылки

Данный интерфейс призван облегчить разработку и использование прикладных программных интерфейсов (API).

PSR-14. Интерфейс простого кэширования

1. Согласно PSR каждый класс создается в отдельном файле

[**https://myrusakov.ru/php-psr-standards.html**](https://myrusakov.ru/php-psr-standards.html)

**Оригинал стандарта** [**https://www.php-fig.org**](https://www.php-fig.org)

**Русский перевод** [**https://svyatoslav.biz/misc/psr\_translation/#\_PSR-1**](https://svyatoslav.biz/misc/psr_translation/#_PSR-1)

**Консоли Windows**

**Microsoft Windows есть 2 терминала:**

**Cmd (для обычных пользователей) и powershell ( для продвинутых пользователей)**

**Cmder**

**Включение настроек Win+Alt+P**

**Cmder -** создан из-за отсутствия хороших эмуляторов консоли в Windows. **https://cmder.net** На оффициальном сайте предлагается две версии: минимальная и полная. В полную версию входит система контроля версий git (Ресурсы для изучения GIT). Cmder не требует установки. Достаточно скачать, разархивировать и запустить файл «cmder.exe».сайт для закачки

[**https://cmder.net**](https://cmder.net)**.**

1. **Команды cmder**
2. **Меню-> При старте ->Запускать выбранную задачу->{bash::bash}**
3. **Идти к «Настройки»Setting выбор интерпретатора оболочки** Win+Alt+P
   1. Настройки командной оболочки Setting->Startup->Task->bash:bash
   2. Перезапускаю Cmder и проверяю, что запустился bash. Ввожу команду echo $SHELL, она отвечает /usr/bin/bash, значит точно баш::баш
4. **Выбор настройки русского языка** Setting->Startup->Environment(Окружение)

set LC\_ALL=ru\_RU.UTF-8

set LANG=ru\_RU.UTF-8

**проверка команда locale**

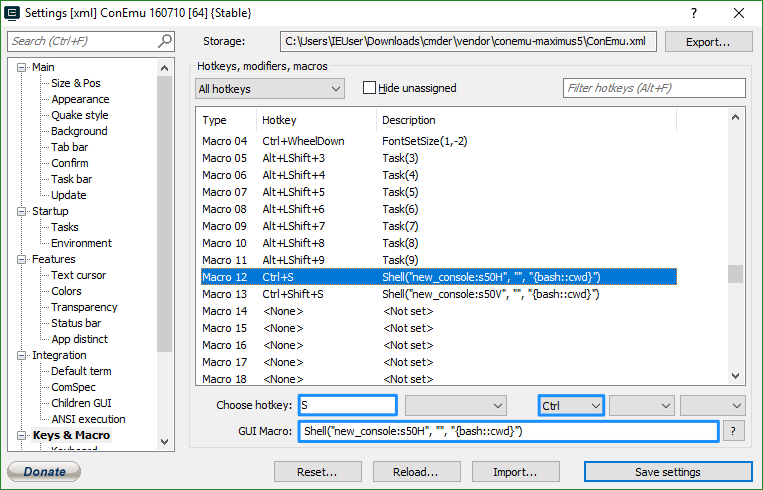
**Настрйока GIT на русский язык >> git config --global core.quotepath false**

* 1. **Файл настроек clink.lua ( в нем невозможно хранить паки и каталоги кирилицей)**

1. **Для выбора настройки открытия рабочего каталога** Setting-Startup->Task->bash:bash->Startup\_dir (**выбрать папку запуска**) (должно появиться запись)-new\_console:d:"E:\Dropbox\Личная моя\domains\СИ\ConsoleApplication1"
2. **Для открытия некольких окон комбинация Ctrl-T**
3. Теперь в настройках Startup → Tasks создам новую задачу. Назову её {bash::cwd}, потому что она запускает bash в текущей папке — current working directory. Поле «Task parameters» скопирую из задачи {bash::bash}, команду тоже копирую из этой задачи, но меняю в последнем параметре папку на %CD%:
   1. cmd /c "%ConEmuDir%\..\git-for-windows\bin\bash --login -i" -new\_console:d:%CD%
   2. В поле Hotkey жму на многоточие и выбираю Ctrl+T: Проверяю — не работает. Открывается дурацкое окно с вопросами. Иду в настройках в «Keys & Macro» и вижу, что на Ctrl+T всё ещё висит «открыть окно с вопросами», выбираю его и добавляю к хоткеям шифт, чтобы окно висело на Ctrl+Shift+T: Ctrl+Shift+T <https://isqua.ru/blog/2016/11/05/nastroika-tierminala-cmder-v-windows/>

Получаю Ctrl+Shift+T -> запуск нового **Cmder c запроса**

Ctrl+T-> запуск нового **Cmder без запросом**

 В меню «Keys & Macro» есть свободные слоты «Macro N», куда можно вписать свои команды и хоткеи. У меня первыми свободными слотами были 12 и 13. В первый я добавил команду: Shell("new\_console:s50H", "", "{bash::cwd}")

s50H — это значит поделить (split) экран на 50% по горизонтали (Horizontal). Добавил хоткей Ctrl+S (от слова Split). В следующий слот добавил деление по вертикали (V): Shell("new\_console:s50V", "", "{bash::cwd}")

Вот хоткеи для вкладок пачкой:

Новая вкладка Ctrl+T

* новая вкладка с настройкой Ctrl+Shift+T
* открыть панель справа Ctrl+S
* открыть панель снизу Ctrl+Shift+S
* закрыть вкладку Ctrl+W.

**Запуск из папки**

Иду в настройки → Integrations. Там две группы полей: «ConEmu Here» и «ConEmu Inside». Первый добавляет в контекстное меню пункт, по которому можно открыть окно ConEmu, а второй добавляет пункт, по которому можно открыть ConEmu прямо во вкладке проводника. Мне нужен первый вариант — открыть ConEmu. Прописываю команду:**{bash::cwd} -cur\_console:n**

1. **ALIAS (Алиасы)** Работают только в оболочке Cmd

Короткое, удобное для запоминания имя, использующееся вместо более длинного и сложного имени; наиболее часто используется в приложениях электронной почты. Также известно как "никнейм". псевдонимы идентификаторов в программировании. **alias** — встроенная команда интерпретаторов командной строки, таких как bash.

Для cmder в режиме cmd находятся в файле **user\_aliases.cmd** папка **E:\Dropbox\cmder\config**

**Для работы bash::bash создаем файл .bashrc в каталоге C:\Users\John**

**Нельзя записывать с пробелом перед и после “=”**

**МОИ АЛИАСЫ**

**alias gs='git status'**

**alias logg='git log --oneline --all --graph'**

**alias la='ls -Al' # показать скрытые файлы**

**alias ls='ls -hF --color' # выделить различные типы файлов цветом**

**alias lx='ls -lXB' # сортировка по расширению**

**alias lk='ls -lSr' # сортировка по размеру**

**alias lc='ls -lcr' # сортировка по времени изменения**

**alias lu='ls -lur' # сортировка по времени последнего обращения**

**alias lr='ls -lR' # рекурсивный обход подкаталогов**

**alias lt='ls -ltr' # сортировка по дате**

**alias lm='ls -al |more' # вывод через 'more'**

**alias tree='tree -Csu' # альтернатива 'ls'**

**# -> Предотвращает случайное удаление файлов.**

**alias mkdir='mkdir -p'**

**# Если в alias стоит пробел в конце перед " ' " то можно вводить продолжение команды**

**# Для вступление изменений в силу, необходимо сохранить alias и перегрузить bash**

**Добавленные git алиасы сохраняются в файл ~/.gitconfig, откуда всегда можно их удалить или отредактировать**

**git config --global alias.st status**

**git config --global alias.co checkout**

**git config --global alias.ci commit**

**git config --global alias.di diff**

**Мелкие хитрости**

**Ctrl + ~** — показать или скрыть консоль

**Ctrl + L** — очистить экран

**Ctrl + U** — полностью убрать всю набранную команду

**Ctrl + R** — поиск по истории команд

**Alt + ←/→** — перемещение курсора по словам набранной команды

Кнопки клавиатуры «стрелка вверх» и «стрелка вниз» — переход по истории введенных команд (удобно для повтора команды с чуть измененными параметрами).

1. Файл настройки E:\Dropbox\Личная моя\domains\cmder 19.06.14 настройки.xml
2. **Настройка под пользователя**
   1. git config --global user.name "John" /\* внесение имени в глобальные настройки \*/
   2. git config --global user.email [jjohn057@gmail.com](mailto:jjohn057@gmail.com) /\* внесение имени в глобальные настройки \*/
   3. git config --list
3. **Добавления удаленного репозитория**
   1. **git remote add origin** [**git@github.com:John057/htmlakacademy.git**](mailto:git@github.com:John057/htmlakacademy.git)
4. **Авторизация из консоли через SSH**
   1. Создание публичного и приватного ключа SSH(папка ~ (пользователи/John/.ssh))
      1. Переходим в каталог репозитория и создаем папку (mkdir .ssh)
      2. Заходим в SSH cd .ssh
      3. Создаем два ключа SSh публичный и приватный в папке .ssh github\_24062019.pub и github\_24062019
         1. ssh-keygen -t rsa -b 4096 -C “[jjohn057@gmail.com](mailto:jjohn057@gmail.com) “ имя ключа вводим обязательно (что бы не запутаться в будущем) пароль пропускаем (ВВОД)
      4. Публичный ключ загружаю на GitHub Personal\_setting-> SSH and GPG keys->копируем содержание ключа из файла **github\_24062019.pub в поле SSH keys**
         1. **Проверяю соединение ssh -T -i ~/.ssh/github\_24062019** [**git@github.com**](mailto:git@github.com) **(файл должен быть приватный ключа SSH)**
      5. Для автоматического соединения необходимо создать файл **config** в папке **C:\Users\John\.sshHost github.com IdentityFile ~/.ssh/my19.06.19**
   2. **Получения всех веток из репозитория GITHUB**
      1. **Netstat** утилита командной строки (**CMD\_Windows**)выводящая на дисплей состояние [TCP](https://ru.wikipedia.org/wiki/TCP)-соединений (как входящих, так и исходящих), таблицы маршрутизации, число сетевых интерфейсов и сетевую статистику по протоколам. Она доступна на всех unix-подобных операционных системах, включая [OS X](https://ru.wikipedia.org/wiki/OS_X), [Linux](https://ru.wikipedia.org/wiki/Linux), [Solaris](https://ru.wikipedia.org/wiki/Solaris), и [BSD](https://ru.wikipedia.org/wiki/BSD), также существует и в основанных на [Windows NT](https://ru.wikipedia.org/wiki/Windows_NT) операционных системах, начиная с [Windows XP](https://ru.wikipedia.org/wiki/Windows_XP) вплоть до [Windows 10](https://ru.wikipedia.org/wiki/Windows_10).[**СМОТРИ EXEL**](языки%20программирования.xlsx)
      2. **C**:\Windows\WinSxS\wow64\_microsoft-windows-tcpip-utility\_31bf3856ad364e35\_10.0.14393.0\_none\_6a90097799f18915

<https://ru.wikipedia.org/wiki/Netstat> **NETSTAT.EXE**

* + 1. [PID](https://ru.wikipedia.org/wiki/Process_ID) — идентификатор [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)). https://ru.wikipedia.org/wiki/Процесс\_(информатика)
  1. **CMD\_windows**

Cmd.exe — интерпретатор командной строки (англ. command line interpreter) для операционных систем OS/2, Windows CE и для семейства операционных систем, базирующихся на Windows NT (англ. Windows NT-based). cmd.exe является аналогом COMMAND.COM, которая используется в семействах MS-DOS и Windows 9x.

* 1. **Windows powershell**

расширяемое средство автоматизации от Microsoft с открытым исходным кодом, состоящее из оболочки с интерфейсом командной строки и сопутствующего языка сценариев. Windows PowerShell построен на базе Microsoft .NET Framework и интегрирован с ним.

* 1. **Bash**

**Bash** (от [англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA" \o "Английский язык) ***B****ourne****a****gain****sh****ell*, [каламбур](https://ru.wikipedia.org/wiki/%D0%9A%D0%B0%D0%BB%D0%B0%D0%BC%D0%B1%D1%83%D1%80" \o "Каламбур) *«Born again» shell* — «возрождённый» shell) — усовершенствованная и модернизированная вариация [командной оболочки](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%B0%D0%BD%D0%B4%D0%BD%D0%B0%D1%8F_%D0%BE%D0%B1%D0%BE%D0%BB%D0%BE%D1%87%D0%BA%D0%B0_UNIX" \o "Командная оболочка UNIX) [Bourne shell](https://ru.wikipedia.org/wiki/Bourne_shell). Одна из наиболее популярных современных разновидностей командной оболочки [UNIX](https://ru.wikipedia.org/wiki/UNIX). Особенно популярна в среде [Linux](https://ru.wikipedia.org/wiki/Linux), где она часто используется в качестве предустановленной командной оболочки.

**TOTAL COMANDER**

Total Commander может запускаться с одним или двумя каталогами в качестве параметров командной строки. При этом указанные каталоги будут открыты, соответственно, в одной или в обеих панелях. Это позволяет вам создавать несколько ярлыков для Total Commander в меню "Пуск", которые запускают Total Commander с различными каталогами или даже с упакованными файлами, открытыми в его панелях. Более того, вместо стандартного **wincmd.ini** можно задать другой INI-файл, что позволит нескольким пользователям на одной машине иметь каждому свои настройки Total Commander.

**Синтаксис:**

totalcmd.exe [/o] [/n] [Диск1:\Каталог1 [Диск2:\Каталог2]] [/i=имя.ini] [/f=имя\_ftp.ini]

или:

totalcmd.exe [/o] [/n] [/L=Диск1:\Каталог1] [/R=Диск2:\Каталог2] [/i=имя.ini] [/f=имя\_ftp.ini]

**Параметры:**

**Диск:\Каталог**Загружает первый каталог, переданный Total Commander, в левую панель, а второй — в правую панель. Имена каталогов, содержащие пробелы, **должны** быть заключены в двойные кавычки! Всегда определяйте полный путь, включая диск! Для открытия сохранённого FTP-подключения задайте в качестве пути **FTPOPEN:имя\_подключения**.

**/O**Если Total Commander уже запущен, этот параметр активизирует его и передаёт путь (пути) из командной строки в эту копию программы (независимо от настройки в [диалоге конфигурации](http://tc-image.3dn.ru/Help/TC/html/dlg_config2.html), позволяющей открывать несколько окон Total Commander).

**/N**В любом случае открывает новое окно Total Commander (независимо от настройки в [диалоге конфигурации](http://tc-image.3dn.ru/Help/TC/html/dlg_config2.html), позволяющей запускать только одну копию Total Commander).

**/L=**Устанавливает путь в левой панели.

**/R=**Устанавливает путь в правой панели.

**/S**Воспринимает переданные каталоги как исходный/целевой вместо левого/правого (для совместного применения с параметром **/O**). Например, каталог, переданный с ключом **/R=** будет воспринят как каталог назначения.

**/S=C**Запускает отдельным процессом инструмент сравнения по содержимому без открытия основного окна TC. Пример: totalcmd.exe /S=C c:\файл1 d:\файл2

**/S=L**Запускает отдельным процессом внутренний просмотрщик, передавая ему файл для просмотра (требуется указывать полный путь).

Принимает дополнительные параметры, отделённые двоеточием. Пример: /S=L:AT1C1250

**A**Использовать шрифт для кодировки ANSI/Windows.

**S**Использовать шрифт для кодировки ASCII/DOS.

**V**Использовать пользовательский шрифт.

**T1..T7**Режим просмотра 1-7 (1: только текст, 2: двоичный, 3: шестнадцатеричный, 4: графика/мультимедиа/LS-плагины, 5: HTML, 6: Unicode, 7: UTF-8).

**C<n>**Кодовая страница, например, C1251 для кириллического текста.

**N**Автовыбор, за исключением графики, мультимедиа и плагинов.

**P<x>**В качестве ПОСЛЕДНЕГО параметра: выбрать плагин. Пример: /S=L:Piclview для использования плагина **ICLView** (указывается имя, отображаемое в заголовке окна просмотрщика).

**/S=P:порт**Запускает сервер для подключения по параллельному или USB-порту (именем порта может быть, например, LPT1 или USB).

**/S=S**Запускает отдельным процессом инструмент синхронизации каталогов без открытия основного окна TC. В параметрах командной строки можно указать либо непосредственно два пути, либо шаблон в форме /S=S:имя\_шаблона

Если параметр начинается со знака равенства "=" (например, /S=S:=имя\_шаблона), сравнение начнётся сразу после открытия диалога. Если знак равенства указан без имени шаблона (т.е. /S=S:=), то для сравнения будут взяты каталоги, переданные через командную строку, и последние использовавшиеся настройки.

**/O=файл\_списка**Только в сочетании с /S=S: передаёт файл списка для функции "Выделенные (в главном окне)". Пример: /O=%F

**/T**Открывает переданные каталоги в новых вкладках. Теперь этот параметр работает также, если Total Commander не был запущен до этого.

**/P=**Устанавливает активную панель при запуске: **/P=L** — левую, **/P=R** — правую. Имеет больший приоритет, чем ключ **ActiveRight** файла [wincmd.ini](http://tc-image.3dn.ru/Help/TC/html/inisettings.html).

**/i=имя.ini**Указывает, что вместо **wincmd.ini**, для сохранения настроек следует использовать другой INI-файл **имя.ini** (см. также: [описание INI-файла](http://tc-image.3dn.ru/Help/TC/html/inisettings.html)).

**/F=имя\_ftp.ini**Указывает, что вместо **wcx\_ftp.ini**, для сохранения настроек встроенного FTP-клиента следует использовать другой INI-файл **имя\_ftp.ini**.

**/d=<x>**Задерживает запуск Total Commander на <x> секунд (предполагается использовать в папке автозагрузки, если есть проблемы с другими программами, запускающимися одновременно с Total Commander).

**/INSTALLDRIVER**Устанавливает драйвер параллельного порта — нужен, только если вы хотите использовать [подключение через параллельный порт](http://tc-image.3dn.ru/Help/TC/html/direct_cable_connection.html) как обычный пользователь, а не как администратор. Требует права администратора (работает только в Windows NT/2000/XP).

**/REMOVEDRIVER**Удаляет драйвер, установленный параметром **/INSTALLDRIVER** (работает только в Windows NT/2000/XP).

**/INSTALLDRIVERQ**Устанавливает драйвер параллельного порта без вывода каких-либо сообщений. Это полезно, например, для использования в удалённых скриптах установки.

**/REMOVEDRIVERQ**Удаляет драйвер параллельного порта без вывода каких-либо сообщений.

Параметры командной строки НЕ чувствительны к регистру.

**Примеры:**

totalcmd.exe c:\windowsВ левой панели открывается c:\windows.

totalcmd.exe d:\data c:\programsВ левой панели открывается d:\data, в правой — c:\programs.

totalcmd.exe /R=d:\dataВ правой панели открывается d:\data.

totalcmd.exe "d:\Письма к Иванову"В левой панели — длинное имя, в двойных кавычках!

totalcmd.exe /R="d:\Письма от Петрова"В правой панели — длинное имя.

totalcmd.exe c:\data\backup.zipВ левой панели открывается содержимое архива backup.zip.

totalcmd.exe /I=Petrov.ini /F=MyFtp.iniДля всех настроек используется Petrov.ini (вместо wincmd.ini), а для FTP — MyFtp.ini.

totalcmd.exe /O /L=c:\ /R="d:\doc"Активизирует уже запущенный Total Commander и устанавливает на левую панель c:\, а на правую панель d:\doc.

totalcmd.exe /O /S /R=c:\путьОткрывает c:\путь в текущей панели НАЗНАЧЕНИЯ запущенной программы. Пояснение: Вследствие применения ключа /S Total Commander воспринимает /L как исходный каталог и /R как каталог назначения вместо левого и правого.

totalcmd.exe /O /T /L=c:\путьОткрывает в левой панели уже запущенного Total Сommander новую вкладку c каталогом, соответствующим установленному пути.

Мои Примеры

start "" "C:\TC\TOTALCMD64.EXE" "FTPOPEN:BEGET" "E:\Dropbox\my\domains\syte" (открытие домена слева и папки справа)

C:\TC файлы настройки FTP соединений wcx\_ftp.ini

**БРАУЗЕРЫ**

**Плагин web-developer сайт разработчика https://chrispederick.com/work/web-developer/**

**Веб-разработчик https://addons.mozilla.org/ru/firefox/addon/web-developer/**

**Cайты для имитации просмотра кода на различных мобильных устройствах**

**<https://www.responsinator.com>**

**При адаптации страницы самое главное не название усройства , а его ширина.**

**Страница должна отображаться без нижней полосы прокрутки**

**СЕТИ HOST VPN**

C:\Windows\System32\drivers\etc host

Сайт для опеределиня IP адресов

**Hosting (Хостинг) Хо́стинг**([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *hosting*) — [услуга](https://ru.wikipedia.org/wiki/%D0%A3%D1%81%D0%BB%D1%83%D0%B3%D0%B0) по предоставлению [ресурсов](https://ru.wikipedia.org/w/index.php?title=%D0%92%D1%8B%D1%87%D0%B8%D1%81%D0%BB%D0%B8%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D1%8B%D0%B5_%D1%80%D0%B5%D1%81%D1%83%D1%80%D1%81%D1%8B&action=edit&redlink=1) для размещения [информации](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D0%B8%D1%8F) на [сервере](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%80%D0%B2%D0%B5%D1%80_(%D0%B0%D0%BF%D0%BF%D0%B0%D1%80%D0%B0%D1%82%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)), постоянно находящемся в сети (обычно [Интернет](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D0%BD%D0%B5%D1%82)). <https://ru.wikipedia.org/wiki/Хостинг>

**Виртуальный хостинг** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *shared hosting*) — вид [хостинга](https://ru.wikipedia.org/wiki/%D0%A5%D0%BE%D1%81%D1%82%D0%B8%D0%BD%D0%B3), при котором множество [веб-сайтов](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%B0%D0%B9%D1%82) расположено на одном [веб-сервере](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80). Это самый экономичный вид хостинга, подходящий для небольших проектов .IP адрес один со множеством других сайтов , расположенных на данном сервере

**VPS** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA" \o "Английский язык) *virtual private server*) или **VDS** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA" \o "Английский язык) *virtual dedicated server*), **виртуальный выделенный сервер** — услуга предоставления в аренду так называемого **виртуального выделенного сервера**. Имеет свой выделенный IP адрес

**Препроцессор**

Препроцессор — это компьютерная программа, принимающая данные на входе и выдающая данные, предназначенные для входа другой программы (например, компилятора). О данных на выходе препроцессора говорят, что они находятся в препроцессированной форме, пригодной для обработки последующими программами (компилятор).

https://ru.wikipedia.org/wiki/Препроцессор

**Задачи с ответами**

**для программирования PHP**

http://php720.com/tasks

**система управления содержимым (CMS),**

Joomla!

Wordpress

**Справочник PHP**

**<http://php.net/manual/ru/function.mysql-connect.php>**

**XSS атаки ( уязвимость)**

**<https://habr.com/ru/post/345494/>**

**https://www.securitylab.ru/analytics/432835.php**

11. **написание программ для андроид**

<https://www.oracle.com/technetwork/java/javase/downloads/jdk11-downloads-5066655.html>

установка среды Java от oracle

<https://developer.android.com/studio/>

андроид студио

**ТЕРМИНЫ**

**Репозиторий, хранилище** — место, где хранятся и поддерживаются какие-либо данные. Чаще всего данные в репозитории хранятся в виде файлов, доступных для дальнейшего распространения по [сети](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_%D1%81%D0%B5%D1%82%D1%8C).

Существуют репозитории для хранения программ, написанных на одном языке (например, [CPAN](https://ru.wikipedia.org/wiki/CPAN) для [Perl](https://ru.wikipedia.org/wiki/Perl)) или предназначенных для одной платформы. Многие современные операционные системы, такие как [OpenSolaris](https://ru.wikipedia.org/wiki/OpenSolaris), [FreeBSD](https://ru.wikipedia.org/wiki/FreeBSD) и большинство дистрибутивов [Linux](https://ru.wikipedia.org/wiki/Linux), имеют официальные репозитории, но также позволяют устанавливать пакеты из других мест. Большинство репозиториев бесплатны, однако некоторые компании предоставляют доступ к собственным репозиториям за платную подписку.

Репозитории используются в [системах управления версиями](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%B2%D0%B5%D1%80%D1%81%D0%B8%D1%8F%D0%BC%D0%B8), в них хранятся все документы вместе с историей их изменения и другой служебной информацией. Русское сообщество [Subversion](https://ru.wikipedia.org/wiki/Subversion) рекомендует использовать вместо термина репозиторий термин [хранилище](https://ru.wikipedia.org/wiki/%D0%A5%D1%80%D0%B0%D0%BD%D0%B8%D0%BB%D0%B8%D1%89%D0%B5_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85)[[1]](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BF%D0%BE%D0%B7%D0%B8%D1%82%D0%BE%D1%80%D0%B8%D0%B9#cite_note-1), поскольку он полностью соответствует как прямому переводу слова «repository», так и его понятию.

Существуют различные автоматизированные системы создания репозиториев. Один из типов репозиториев: хранилища на [CD](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D0%B0%D0%BA%D1%82-%D0%B4%D0%B8%D1%81%D0%BA)/[DVD](https://ru.wikipedia.org/wiki/DVD) — установочные диски для пакетов того или иного [ПО](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)

Валидность-Итак, валидность кода — это соответствие исходного кода сайта нормам и правилам, описанным Консоциумом Всемирной Паутины или сокращенно W3C. Чтобы проверить свой блог на соответствие этим нормам нужно пройти по ссылке: validator.w3.org. Ввести нужный адрес и посмотреть результаты.

**API (application programming interface)**

**API** (**программный интерфейс приложения, интерфейс**[**прикладного программирования**](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)). описание способов (набор [классов](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)), [процедур](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D0%B4%D1%83%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)), [функций](https://ru.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%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)), [структур](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%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)) или [констант](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D1%81%D1%82%D0%B0%D0%BD%D1%82%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))), которыми одна компьютерная программа может взаимодействовать с другой программой.

Часто реализуется отдельной программной библиотекой или сервисом операционной системы. Используется программистами при написании всевозможных приложений. API могут быть процедурными или объектно-ориентированными. При использовании процедурных API вы вызываете функции для выполнения каких-либо операций, а в случае объектно-ориентированных вы инстанцируете классы и затем вызываете методы созданных объектов. Второй подход, обычно, предпочтительнее, так как он более современный и способствует написанию более организованного кода.

<https://ru.wikipedia.org/wiki/API>

**(GUI , [англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA" \o "Английский язык) graphical user interface )**

**Графи́ческий интерфе́йс по́льзователя (ГИП)**, **графический пользовательский интерфейс (ГПИ)** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *graphical user interface,****GUI***) — разновидность пользовательского интерфейса, в котором [элементы интерфейса](https://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%B5%D0%BC%D0%B5%D0%BD%D1%82_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81%D0%B0) (меню, кнопки, значки, списки и т. п.), представленные пользователю на [дисплее](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D1%81%D0%BF%D0%BB%D0%B5%D0%B9), исполнены в виде [графических](https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84%D0%B8%D0%BA%D0%B0) изображений. Также называется графической оболочкой управления.

В отличие от [интерфейса 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), в GUI пользователь имеет произвольный доступ (с помощью [устройств ввода](https://ru.wikipedia.org/wiki/%D0%A3%D1%81%D1%82%D1%80%D0%BE%D0%B9%D1%81%D1%82%D0%B2%D0%BE_%D0%B2%D0%B2%D0%BE%D0%B4%D0%B0) — клавиатуры, мыши, джойстика и т. п.) ко всем видимым экранным объектам (элементам интерфейса) и осуществляет непосредственное манипулирование ими. Чаще всего элементы интерфейса в GUI реализованы на основе [метафор](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%B0%D1%84%D0%BE%D1%80%D0%B0) и отображают их назначение и свойства, что облегчает понимание и освоение программ неподготовленными [пользователями](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8C).

**Псевдоклассы**

**Псевдоклассы** определяют динамическое состояние элементов, которое изменяется с помощью действий пользователя, а также положение в дереве документа. Примером такого состояния служит текстовая ссылка, которая меняет свой цвет при наведении на неё курсора мыши. При использовании псевдоклассов браузер не перегружает текущий документ, поэтому с помощью псевдоклассов можно получить разные динамические эффекты на странице.

Синтаксис применения псевдоклассов следующий.

Селектор:Псевдокласс { Описание правил стиля }

<http://htmlbook.ru:8080/samcss/psevdoklassy>

**Рефа́кторинг**

**Рефа́кторинг** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *refactoring*), или **перепроектирование кода, переработка кода, равносильное преобразование алгоритмов** — процесс изменения внутренней структуры [программы](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%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0), не затрагивающий её внешнего поведения и имеющий целью облегчить понимание её работы[[1]](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D1%84%D0%B0%D0%BA%D1%82%D0%BE%D1%80%D0%B8%D0%BD%D0%B3#cite_note-1)[[2]](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D1%84%D0%B0%D0%BA%D1%82%D0%BE%D1%80%D0%B8%D0%BD%D0%B3#cite_note-_41d0e9d6b7e34a0a-2). В основе рефакторинга лежит последовательность небольших эквивалентных (то есть сохраняющих поведение) преобразований. Поскольку каждое преобразование маленькое, программисту легче проследить за его правильностью, и в то же время вся последовательность может привести к существенной перестройке программы и улучшению её согласованности и чёткости.

**Drag-and-drop**

(в переводе с [английского](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) означает буквально *тащи-и-бросай*; **Бери-и-Брось**) — способ оперирования [элементами интерфейса](https://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%B5%D0%BC%D0%B5%D0%BD%D1%82_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81%D0%B0) в [интерфейсах пользователя](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%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F) (как [графическим](https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F), так и [текстовым](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BA%D1%81%D1%82%D0%BE%D0%B2%D1%8B%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8F), где элементы GUI реализованы при помощи [псевдографики](https://ru.wikipedia.org/wiki/%D0%9F%D1%81%D0%B5%D0%B2%D0%B4%D0%BE%D0%B3%D1%80%D0%B0%D1%84%D0%B8%D0%BA%D0%B0)) при помощи манипулятора «[мышь](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%BC%D1%8B%D1%88%D1%8C)» или [сенсорного экрана](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D0%BD%D1%81%D0%BE%D1%80%D0%BD%D1%8B%D0%B9_%D1%8D%D0%BA%D1%80%D0%B0%D0%BD).

Способ реализуется путём «захвата» (нажатием и удержанием главной (*первой*, чаще левой) кнопки мыши) отображаемого на [экране](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D1%81%D0%BF%D0%BB%D0%B5%D0%B9) [компьютера](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80) объекта, программно доступного для подобной операции, и перемещении его в другое место (для изменения расположения) либо «бросания» его на другой элемент (для вызова соответствующего, предусмотренного программой, действия). По отношению к [окнам](https://ru.wikipedia.org/wiki/%D0%9E%D0%BA%D0%BD%D0%BE_(%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)) (также способным к перемещению подобным способом) данный термин обычно не употребляется.

**Вопросы**

**Среда для установки**

1. PHPStorm J:\Программирование Основы\Прога
2. Notepad++
3. GIT пароли в данном файле
4. GITHUB пароли в данном файле GUI GitHubDesktopSetup.exe
5. OpenServer J:\Программирование Основы\Прога\Open Server phpMyAdmin

**Java**

**Java платформа**

Скачиваем с официального сайта <https://www.oracle.com/technetwork/java/javase/downloads/index.html>

jdk-11.0.2\_windows-x64\_bin.exe

запускаем. Распаковывается автоматически в каталог C:\Program Files\Java\jdk-11.0.2

После этого обязательно перегрузить комп.

**Проверка установки осуществляется программа установка!!!**

**PHPStorm**

**[https://www.jetbrains.com/products.html?fromMenu#type=language](https://www.jetbrains.com/products.html?fromMenu" \l "type=language) сайт разработчика**

скачал из rutracker.org

после установки запускаем файл лечения JetBrains\_KeyfileMaker.jar выбираем PHPStorm и генерируем файл phpstorm.key , скопируем его в C:\Users\Евгений\.PhpStorm2018.1\config. ВСЕ РАБОТАЕТ!!!

Скачиваю с rutracker.

Устанавливаю и запускаю первый раз.

запускаю keygey JetBrains\_KeyfileMaker.jar

Jar ( это архив Java script) выбираю PHPStorm и копию сгенерированный файл phpstorm.key в C:\Users\Евгений\.PhpStorm2018.1\config

PhpStorm — коммерческая кросс-платформенная интегрированная среда разработки для PHP[3]. Разрабатывается компанией JetBrains на основе платформы IntelliJ IDEA.

PhpStorm представляет собой интеллектуальный[4] редактор для PHP, HTML и JavaScript с возможностями анализа кода на лету, предотвращения ошибок в коде и автоматизированными средствами рефакторинга для PHP и JavaScript. Автодополнение кода в PhpStorm поддерживает спецификацию PHP 5.3, 5.4, 5.5, 5.6, 7.0, 7.1 и 7.2 (современные и традиционные проекты), включая генераторы, сопрограммы, пространства имен, замыкания, типажи и синтаксис коротких массивов. Имеется полноценный **SQL-редактор** с возможностью редактирования полученных результатов запросов

**Плагин EMMET**

[**http://emmet.io**](http://emmet.io)

**https://dwstroy.ru/stail/plaginy-rasshireniya/emmet-shpargalka/**

**Сниппет snippet**

**(**англ. snippet — отрывок, фрагмент) — фрагмент исходного текста или кода программы, применяемый в поисковых системах, текстовых редакторах и средах разработки.

**NOTEPAD++**

Устанавливать только 32 bit версию

**Установка для работы с FTP**

**-Добавления менеджера плагинов и плагина NppFTP**

**Вкладка ПЛАГИНЫ -> Управление Плагинамси -> NppFTP -> УСТАНАВЛИВАЕМ ->**

**Настраиваем BEGET**

Скачать plugin manager: <https://github.com/bruderstein/nppPlu>...

Скачал в папку NOTEPAD

Скопировать **PluginManager.dll** в папку **C:\Program Files (x86)\Notepad++\pluginsPlugin**

Скопировать **gpup.exe** в папку C:\Program Files (x86)\Notepad++\updater

Должен появиться в меню **ПЛАГИНЫ**

**Настройка на правильную работу \*.bat файлов**

**Для правильного восприятия Русского текста в именах файлах необходимо**

**Кодировка-> Кодировка -> кирилица -> OEM866**

**Подключить интерпретор CMD (аналогично браузерам) ALT-> F12**

* **Подключение Браузеров**
  + **Вкладка Запуск -> Изменение сочетане клавиш -> удаляем все запуски(кроме браузеров)**

1. Iexplorer Назв. **IE** сочетание **ALT->F1** команда **"C:\Program Files (x86)\Internet Explorer\iexplore.exe" "$(FULL\_CURRENT\_PATH)"**
2. **Opera Назв. Opera сочетание ALT->F2 команда C:\Users\John\AppData\Local\Programs\Opera\launcher.exe "$(FULL\_CURRENT\_PATH)"**
3. **Firefox Назв. Firefox сочетание ALT->F3 команда "C:\Program Files\Mozilla Firefox\firefox.exe" "$(FULL\_CURRENT\_PATH)"**
4. **H:\Программирование Основы\Прога\NOTEPAD ++ Браузеры подключение и запуск.pdf**

**Как запустить внешнюю программу**

**и**спользование переменной окружения: $(ПЕРЕМЕННАЯ\_ОКРУЖЕНИЯ)  
Например, текущий файл, который вы редактируете в Notepad++ — «E:\my Web\main\welcome.html».  
Ниже приведены некоторые переменные окружения, содержащие информацию о текущей сессии:  
**FULL\_CURRENT\_PATH** — «E:\my Web\main\welcome.html».  
**CURRENT\_DIRECTORY** — «E:\my Web\main\».  
**FILE\_NAME** — «welcome.html».  
**NAME\_PART** — «welcome».  
**EXT\_PART** — «html».  
**SYS.<переменная>** — имя системной переменной окружения, например, $(SYS.PATH).  
Переменные стоит обрамлять двойными кавычками, поскольку они могут содержать пробелы:  
"$(ПЕРЕМЕННАЯ\_ОКРУЖЕНИЯ)"  
Примеры  
firefox "$(FULL\_CURRENT\_PATH)"  
iexplore "$(FULL\_CURRENT\_PATH)"

**Как стать начинающим верстальщиком**

http://glivera-team.github.io/how-to/2016/05/02/how-to-start.html

02.05.2016

Привет, читатель!

В этой статье я расскажу как стать начинающим верстальщиком. Т.е. после выполнения всех рекомендаций в можете претендовать на данную позицию. Статья создана специально для учеников glivera-team, но может помочь всем, кто решил освоить данную специальность.

**Учим HTML/CSS**

HTML и CSS - это то, что верстальщик пишет каждый день, это языки, которые понимает браузер и рисует ваш сайт. Вы можете выбрать любой источник для изучения(книги, видео, статьи), главное понять основы. Вот несколько полезных ссылок, которые помогут начать:

* [HTML для начинающих от loftblog(видео-уроки)](https://www.youtube.com/playlist?list=PLY4rE9dstrJyeZlPWoKJr1xKVVnG4w-Hc)
* [Основы CSS от Sorax(видео-уроки)](https://www.youtube.com/playlist?list=PL026CCEB5125879C2)
* [Интерактивные курсы от HTML Academy](https://htmlacademy.ru/)
* [Визуальное руководство по свойствам Flexbox](http://css-live.ru/articles/vizualnoe-rukovodstvo-po-svojstvam-flexbox-iz-css3.html)
* [Flexbox CSS3(видео-уроки)](https://www.youtube.com/watch?v=RNjnRA0QSug)

После прохождения уроков вы должны понимать:

* что такое html-теги и как они работают
* блочную модель(border, margin, padding)
* как работает float, inline-block и flexbox
* позиционирование(абсолютное, относительное, фиксированное, z-index)

Вы должны уметь:

* вставить на страницу текст, картинки, ссылки, элементы форм
* стилизовать элементы страницы(изменение размеров, цветов, теней и т.д)
* создавать таблицы

**Работаем с фотошопом**

Дизайнер предоставляет верстальщику макет в формате psd(формат файлов Photoshop). Поэтому верстальщику необходимо уметь с ним работать. Все функции программы не нужны, главное уметь следующее:

* работа со слоями(скрытие, отображение, просмотр эффектов)
* замер размеров элементов
* вырезание картинок(обычных и паттернов)
* копирование текста из макета в html

Полезные ссылки:

* [Фотошоп для верстальщика(хабр)](https://habrahabr.ru/post/272649/)
* [С ножом против паровоза, В. Макеев](https://www.youtube.com/watch?v=9lzJvnJpXL8)
* [Фотошоп для кодера(цикл статей)](http://xiper.net/learn/photoshop/)
* [Основы Photoshop для web разработчика](https://www.youtube.com/watch?v=Kmkx3SfJwuc)

**Начальные инструменты**

Для быстрого старта вам понадобится редактор и браузер. Вы можете выбрать любые, но я посоветую Sublime Text в качестве редактора и Chrome в качестве браузера. Из операционных систем сделайте выбор между Windows и Mac. Остальные системы будут тормозить ваше обучение.

**Верстка первого сайта**

На этом этапе скорее всего вы ощутите ступор. Вы вроде знаете html/css(на самом деле нет) и фотошоп. Но вы совершенно не понимаете как из макета сделать сайт. В данном случае вы можете пройти небольшой курс по верстке сайта, коих много на ютубе. На данном этапе вам нужно сверстать первый сайт. Не обращайте внимание на такие понятия как резина, адаптивность. И еще, не используйте css-фреймворки(bootstrap, foundation).

Полезные ссылки:

* [Курс видеоуроков “Джедай верстки”](https://www.youtube.com/watch?v=Xi0S4xOS8HY&list=PLyf8LgkO_8q9BWUTfXtH3PQnPl1zgkOX7)
* [Как сверстать сайт с нуля(видеоуроки)](https://www.youtube.com/watch?v=vFjRVugE030)

**Понимание семантики, валидность**

Итак, вы сверстали первый сайт, что дальше? Вам нужно понять как правильно расставлять теги в html-коде, а также научиться проводить [валидацию](https://validator.w3.org/). С валидацией все просто - машина анализирует код, говорит что не так, вы правите и понимаете. С семантикой посложнее, поскольку область довольно холиварная и во многих моментах разработчики не имеют единого мнения. Но все же общие стандарты есть. После того, как изучите - попробуйте сверстать новый макет семантично. Теперь на каждом этапе вам нужно получить обзор вашего кода от опытного специалиста/наставника.

Полезные ссылки:

* [Искусство семантики HTML(статья)](http://frontender.info/the-art-of-html-semantics-pt1/#iskusstvosemantikihtmlchasty1)
* [Вёрстка со смыслом. Новая семантика HTML5(видео)](https://www.youtube.com/watch?v=KKDC3emgfaU)

**Javascript и jQuery**

На этом этапе вы должны понять, что такое Javascript и зачем он нужен. Далее познакомиться с библиотекой jQuery и попробовать сверстать сайт с интерактивными элементами(слайдеры, всплывающие окна, табы).

Полезные ссылки:

* [Основы JS от Sorax](https://www.youtube.com/playlist?list=PL363QX7S8MfSxcHzvkNEqMYbOyhLeWwem) (аккуратнее, может взорваться голова)
* [Основы jQuery от Loftblog](https://www.youtube.com/playlist?list=PLY4rE9dstrJxST3xftg5kSCQdUB8hvj4S)
* [Базовые элементы страниц от Loftblog](https://www.youtube.com/playlist?list=PLY4rE9dstrJz25gq63HjXIOPg0ailynCN)
* [Русская документация по API jQuery](https://jquery-docs.ru/)

**БЭМ**

Изучите методологию БЭМ. Вам нужно понимать, как верстать независимыми блоками, как привнести модульность в вашу верстку. В интернете очень много информации по этому поводу. Не нужно изучать фул-стек БЕМ, просто важно понять методологию для CSS. Верстаете новый макет с применением методологий.

Полезные ссылки:

* [Как в glivera-team применяют БЭМ(статья)](http://glivera-team.github.io/structure/2016/02/11/kbem.html)
* [Пишем БЭМ правильно(видео)](https://www.youtube.com/watch?v=hTmxbJF2Tts)
* [BEM (БЭМ) & SMACSS — Sass методологии для организации проектов(видео)](https://www.youtube.com/watch?v=vXW7w3ym8hg)

**SASS**

Для ускорения и удобства написания стилей придумали препроцессоры. Наш выбор - Sass(.scss). Вам необходимо изучить препроцессор и сверстать с ним новый макет. Понять нужно:

* переменные
* разница между миксинами и тихими классами(placeholder)
* как работает ‘&’
* как разделить стили на несколько .scss файлов

На данном этапе scss в css компилируйте с помощью [prepros](https://prepros.io/)

Полезные ссылки:

* [Все про Sass и Compass(видео)](https://www.youtube.com/playlist?list=PLZfRjCZl2NuQr8v2_DV8ZX6a03gntn7yU)
* [sass-lessons.ru](http://sass-lessons.ru/)

**Pug**

Также для ускорения верстки и поддержки верстальщики используют шаблонизаторы html. Наш выбор - pug. После изучения вы должны понимать:

* как делать миксины
* как работает extends
* циклы и переменные
* работа с массивами

Верстаете сайт с применением шаблонизатора.

Полезные ссылки:

* [Видео про Jade(Pug) на loftblog](https://www.youtube.com/watch?v=dnL5EpZWdgw)
* [https://pugjs.org](https://pugjs.org/)
* [Туториал по HTML препроцессору Pug (Jade)](https://gist.github.com/neretin-trike/53aff5afb76153f050c958b82abd9228)

**Адаптивность**

Теперь вы можете начать верстать адаптивные сайты, которые подстраиваются под ширину окна браузера. Можете взять макет, где различные разрешения указаны дизайнером, а можете сами придумать, как будет перестраиваться сайт. С этого момента все сайты верстайте адаптивно.

Полезные ссылки:

* [Видеокурс по адаптивным сайтам](https://www.youtube.com/watch?v=Au6d1d-vFjU&list=PLY4rE9dstrJyJEghRkl7qfqdAdlyvaOVQ)

**IDE PHPStorm**

Чем раньше вы пересядете с редактора на PHPStorm - тем лучше. Эта IDE позволяет значительно ускорить разработку. Скачать бесплатную(early access) PHPStrom можно на [официальном сайте](https://confluence.jetbrains.com/display/PhpStorm/PhpStorm+Early+Access+Program), если не хотите платить - скачиваете её раз в месяц. В остальном бесплатная версия такая же, как и платная.

Полезные ссылки:

* [Настройка PhpStorm для вёрстки на ОС Windows(хабр)](https://habrahabr.ru/post/282003/)
* [PhpStorm Video Tutorials](https://www.youtube.com/playlist?list=PLQ176FUIyIUbfeFz-2EbDzwExRlD0Bc-w)

**Git**

Любой разработчик должен уметь работать с системами контроля версий, и верстальщик - не исключение. Мы, как и большинство выбрали Git. Вы должны уметь следующее:

* работать с интерфейсом github
* уметь клонировать репозитории на компьютер
* делать commit, push, merge
* делать pull request

Полезные ссылки:

[Git для новичков(видео)](https://www.youtube.com/playlist?list=PLY4rE9dstrJyTdVJpv7FibSaXB4BHPInb)

[Шпаргалка по git](http://dev-lab.info/2013/08/%D1%88%D0%BF%D0%B0%D1%80%D0%B3%D0%B0%D0%BB%D0%BA%D0%B0-%D0%BF%D0%BE-git-%D0%BE%D1%81%D0%BD%D0%BE%D0%B2%D0%BD%D1%8B%D0%B5-%D0%BA%D0%BE%D0%BC%D0%B0%D0%BD%D0%B4%D1%8B-%D1%81%D0%BB%D0%B8%D1%8F%D0%BD/)

**Практика**

Дальше вам необходимо верстать и набивать руку, каждый макет прогоняя через код-ревью и найти первый заказ. Затем второй, третий и т.д.

На этом все.

**Сети VPN DDNS**

**VVPN (англ. Virtual Private Network ) – виртуальная частная сеть**

**Домашний VPN предоставляет вам зашифрованный туннель – соединение, позволяющее вам использовать открытые публичные сети Wi-Fi для создания подключения к нужной VPN –сети.**

**динамической системы доменных имен (DDNS)**

**Функция "Динамический DNS" (Dynamic DNS) позволяет присвоить постоянное доменное имя (адрес для доступа из интернета) публичному, динамическому IP-адресу, который роутер получает от провайдера**.

[**https://2ip.ru**](https://2ip.ru)

**46.147.99.26 25/03/2020**

Белый, публичный IP-адрес (даже если он динамический) позволяет получить доступ к нашему роутеру из интернета. То есть, этот адрес обеспечивает прямую связь из сети интернет и маршрутизатором.

Так как белых IP-адресов на все устройства в сети интернет не хватает (возможно, это не основная причина), то интернет-провайдеры очень часто выдают своим клиента (роутерам, устройствам) **серые IP-адреса (они же приватные, или частные).** Это адреса из локальной (частной) сети интернет-провайдера. А уже из локальной сети провайдера идет выход в интернет через один внешний IP-адрес. Он может быть общим для определенного количества клиентов.

**функция DDNS через серый IP-адрес работать не будет. Чтобы настроить динамический DNS, необходимо, чтобы провайдер выдавал вам белый, публичный IP-адрес.**

**Опеределение Белого или Серого IP адреса**

* 1. **Зайти в настройки роутера и узнать внешний IP адрес (пример 100.104.144.16**)
  2. **Зайти на сайт** [**https://2ip.ru**](https://2ip.ru) **и узнать свой IP адрес 46.147.99.26**
  3. **Если не совпадет значит вы подключены через сеть провайдера**

**Протоколы**

**SSH (Secure Shell — «безопасная оболочка»)**

**SSH (**Secure Shell — «безопасная оболочка») —сетевой протокол прикладного уровня, позволяющий производить удалённое управление операционной системой и туннелирование TCP-соединений (например, для передачи файлов). Схож по функциональности с протоколами Telnet и rlogin, но, в отличие от них, шифрует весь трафик, включая и передаваемые пароли. SSH допускает выбор различных алгоритмов шифрования. SSH-клиенты и SSH-серверы доступны для большинства сетевых операционных систем.

SSH позволяет безопасно передавать в незащищённой среде практически любой другой сетевой протокол.**.**

[**https://www.opennet.ru/cgi-bin/opennet/man.cgi?topic=ssh&category=1**](https://www.opennet.ru/cgi-bin/opennet/man.cgi?topic=ssh&category=1) **протокол подключения**

**Отличие HTTP от HTTPS**

HTTPS (аббр. от [англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) HyperText Transfer Protocol Secure) — расширение [протокола](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%82%D0%BE%D0%BA%D0%BE%D0%BB_%D0%BF%D0%B5%D1%80%D0%B5%D0%B4%D0%B0%D1%87%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) [HTTP](https://ru.wikipedia.org/wiki/HTTP) для поддержки [шифрования](https://ru.wikipedia.org/wiki/%D0%A8%D0%B8%D1%84%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) в целях повышения безопасности. Данные в протоколе HTTPS передаются поверх криптографических протоколов [TLS](https://ru.wikipedia.org/wiki/TLS) или устаревшего в 2015 году [SSL](https://ru.wikipedia.org/wiki/SSL)[[1]](https://ru.wikipedia.org/wiki/HTTPS#cite_note-1). В отличие от HTTP с TCP-портом 80, для HTTPS по умолчанию используется [TCP](https://ru.wikipedia.org/wiki/TCP)-[порт](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D1%80%D1%82_(TCP/IP)) 443[[2]](https://ru.wikipedia.org/wiki/HTTPS#cite_note-%D0%B0%D0%B2%D1%82%D0%BE%D1%81%D1%81%D1%8B%D0%BB%D0%BA%D0%B01-2).

HTTPS не является отдельным протоколом. Это обычный HTTP, работающий через шифрованные транспортные механизмы [SSL](https://ru.wikipedia.org/wiki/SSL) и [TLS](https://ru.wikipedia.org/wiki/TLS)[[4]](https://ru.wikipedia.org/wiki/HTTPS#cite_note-4). Он обеспечивает защиту от атак, основанных на прослушивании сетевого соединения — от [снифферских](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%82%D0%BE%D1%80_%D1%82%D1%80%D0%B0%D1%84%D0%B8%D0%BA%D0%B0) атак и атак типа [man-in-the-middle](https://ru.wikipedia.org/wiki/%D0%A7%D0%B5%D0%BB%D0%BE%D0%B2%D0%B5%D0%BA_%D0%BF%D0%BE%D1%81%D0%B5%D1%80%D0%B5%D0%B4%D0%B8%D0%BD%D0%B5), при условии, что будут использоваться шифрующие средства и сертификат сервера проверен и ему доверяют[[5]](https://ru.wikipedia.org/wiki/HTTPS#cite_note-5).

**https://ru.wikipedia.org/wiki/HTTPS**

**Контакты WEB разработчиков**

Sorax

Группа ВК: http://vk.com/soraxcss

Я ВК: http://vk.com/art.sorax

Я на FB: http://www.fb.com/art.sorax

Я на Formspring: http://www.formspring.me/artsorax

loftblog

Telegram: https://telegram.me/loftblog

Slack: http://slack.loftblog.ru/

Наш сайт: http://loftblog.ru/

Instagram: https://www.instagram.com/loftblog/

ВКонтакте: http://vk.com/loftblog

facebook: http://www.facebook.com/loftblog

twitter: http://twitter.com/loft\_blog