

PHP для начинающих программистов

Длительность курса: 17 академических часов

1 Базовые знания

- | | | |
|---|-----------------------------------|---|
| 1 | Введение и базовые понятия | узнать, как работает сайт с применением PHP —
разобраться, как приходит запрос, как он передаётся веб-серверу, как он передаётся PHP на выполнение.
Установить сборку OpenServer / XAMPP
Установить ПО для написания кода. |
|---|-----------------------------------|---|
-

2 Переменные, типы	Понять, что такое переменные. Познакомиться с типами переменных. Узнать про приведение типов и их преобразование. Узнать про правильное именование переменных. Реализовать простые математические операции. Константы. Научиться использовать комментарии
3 Ветвления	Познакомиться с блок-схемами и тем, как они описывают различные логические схемы Познакомиться с конструкциями if, if-else, if-else if-else. В качестве условий можно брать математические операции сравнения. Познакомиться с конструкцией switch Познакомиться с тернарным оператором. Чем тернарный оператор в PHP 7 отличается от него же, но в PHP 5? Узнать, какие условия возможны в PHP. Что такое empty, isset, !empty ? Что такое «==» и «===»? Чем они отличаются?
4 Функции	Познакомиться с функциями Посмотреть, какие переменные и в каком количестве можно передавать в функции Что такое возвращаемое значение? Как оно происходит? Узнать, что такое typehinting в PHP7 Что такое передача аргументов по ссылке? Что такое статические и глобальные переменные? Чем плохи глобальные переменные? Узнать про правильное именование функций Узнать и научиться правильно применять рекурсию (классическая задача — вычисление ряда Фибоначчи или возведение числа в степень). При изучении массивов нужно будет вернуться к вопросам рекурсии Изучить синтаксис комментариев в PHPDoc.

5 Циклы и массивы

Узнать, что такое массив.

Познакомиться с индексными и ассоциативными массивами.

Узнать, что такое многомерные массивы.

Слить два массива при помощи `array_merge`.

Что такое `array_shift`, `array_pop`, `explode`, `implode`?

Познакомиться с циклами `while`, `do-while`, `for`. Понять, что `for` вполне достаточно (нужно научиться заменять `while` и `do-while` при помощи `for`).

Вывести на экран разные циклы

Узнать, что такое бесконечный цикл, а также конструкции `break` и `continue`.

Узнать, что такое цикл `foreach`.

Создать двумерный массив, обойти его вложенными циклами."Узнать, что такое массив.

Познакомиться с индексными и ассоциативными массивами.

1 Работа с файлами

Поработать с простыми файлами — почитать и позаписывать данные.

Научиться записывать в файл логи выполнения программы (записать в файл содержимое переменной). Какие режимы работы с файлами существуют?

Научиться работать со структурированной информацией — CSV-форматом.

Научиться получать списки файлов в директориях и сформировать фотогалерею из файлов, лежащих в указанной папке.

Узнать про разницу между абсолютными и относительными адресами.

Узнать, что такое домашняя директория сайта, домашняя директория хоста, корневая директория.

Чем они отличаются? Когда они нужны?

Как подключать файлы с кодом? В чём разница между `include` и `require`? Зачем им `_once`?

2 Интерактивность

Узнать про методы HTTP (GET, PUT, POST, DELETE и т.д.) Чем они отличаются? Чем HTTPS отличается от HTTP? Что такое заголовки? Какие статусы ответа может возвращать сервер?

Посмотреть, как создать форму на HTML, которая будет отправлять данные к скрипту `script.php`

Узнать, что такое суперглобальные массивы `$_REQUEST`, `$_POST`, `$_GET`, `$_FILE`

Научиться загружать файл на сервер.

Улучшить фотогалерею — теперь человек сам загружает файлы изображений.

3 **Работа с базами данных**

Научиться подключаться к ней при помощи клиента. Например, HeidiSQL.

Научиться создавать простые таблицы с данными.

Изучить простейшие команды выполнения запросов — SELECT, INSERT, UPDATE, DELETE.

Усовершенствовать обработку формы обратной связи — сохранять полученные данные в БД.

Узнать, что такое SQL-инъекция. Как её избежать?

4 **Сессии и Cookie**

Познакомиться с механизмом сессий. Выяснить, сколько хранится сессия, как ею управлять.

Создать форму авторизации. Логика простая — человек вводит пару логин пароль. После этого его состояние можно сохранять в базе данных и в сессиях.

Узнать, почему нельзя хранить пароль пользователя в открытом виде. Узнать, что такое хэш (md5, blowfish, crc32, base64).

Научиться сохранять в сессии данные о пользователе после авторизации. Например, создать два скрипта PHP, и показывать на них имя пользователя в случае, если пользователь авторизован.

Познакомиться с механизмом Cookie. Чётко понимать, что можно хранить в них, а что нельзя.

Почему Cookie — небезопасное хранилище?

Реализовать функционал «Запомнить меня» в форме логина

Узнать, как защититься от XSS, CSRF атак. Почему и как нужно защищать формы?

1 ООП

Выяснить, как создаются классы и объекты.
Разобраться с состоянием объекта, передачей по ссылке. Свойства объекта как отражение состояния. Статические свойства и методы — когда они нужны, а когда нет.

Основные принципы ООП: полиморфизм, инкапсуляция и наследования — что это такое и как реализуется в PHP

Ключевые слова `static`, `parent`, `self`, `this`

Тайпхинтинг в свойствах и методах (PHP 7+)

Понять, чем отличаются абстрактные классы и интерфейсы.

2 MVC.
Composer.
Автозагрузка

Изучить подход. Как реализуется MVC в PHP? Что такое пассивный и активный MVC? Узнать про «уродливый жирный контроллер»

Front Controller или единая точка входа для реализации MVC, работа с приложением через инстанс (объект) самого приложения

Структура директорий MVC-проекта

Роутинг в MVC — как обрабатывать и

конфигурировать преобразование адресов в команду приложению

Создать самое простое MVC-приложение:

отображение 2-3 страниц при помощи единой точки входа и MVC.

3 PDO

PDO — создание соединения, запросы, `prepare`

Кто должен запрашивать данные: модель или контроллер?

Разделение бизнес-логики в моделях и кода отражения сущностей в БД.

- 1 **Шаблонизатор Twig**

Структурировать работу с шаблонами
Научиться пользоваться более продвинутым шаблонизатором
Узнать о возможностях обработки данных на уровне шаблонов.

- 2 **Инструменты разработки**

Установить Xdebug и попробовать создать ситуацию с ошибкой
Установить Xdebug и попробовать создать ситуацию с ошибкой
Познакомиться с библиотекой Monolog
Конфигурирование приложения — уберите хардкод (заодно узнаете, что это) и настройки приложения в выделенные файлы. Плейсхолдеры в коде
Механизмы конфигурирования в php — ini-файлы, yaml, xml, json
Механизм исключений — что такое Exception, try-catch-finally, наследование исключений и их проброс.

- 3 **Кэширование**

Когда нужен кэш? БД, API, файловая система. Кэш есть везде. Он должен быть и в Вашем приложении.
Механизмы кэширования: OPCache, Redis, Memcached
Что кэшировать и на какой срок? Страницы, блоки данных, значения переменных, результаты запросов
Прогрев кэша
Интеграция Memcached и Nginx.

1 Практикум

Пишем наше первое полноценное приложение