1. Общая компьютерная грамотность
   1. Файловая система, что такое файл, папка, путь
   2. Основные команды консоли
   3. Процессы, запуск и остановка
   4. Знакомство с \*nix OS
2. Понятия об этапах обработки запроса

Нельзя не знать о том, как работает протокол HTTP

1. HTML и CSS
2. Bootstap
3. DOM
4. JavaScript, JQuery

PHP

1. Система типов
2. Какие типы существуют в PHP?
3. Динамическая нестрогая типизация
4. Тип имеет значение, а не переменная
5. Правила приведения типов (основы)
6. Все суть выражение
7. Операторы арифметические и др.
8. Логические операции
9. Приоритет операторов
10. Присваивание
11. Ссылки
12. Переменные и константы
13. Условия, циклы
14. Внешние данные
15. $\_GET, $\_POST
16. $\_FILES
17. Include и require
18. Функции
19. Сто функций стандартной библиотеки
20. Аргументы и возврат значения, генерация значений
21. Анонимные функции, функции как объекты первого класса
22. Области видимости, локальность
23. Объектная модель
24. Классы и объекты. Инкапсуляция, наследование, полиформизм
25. Конструкторы
26. Магия в ООП
27. Статика, LSB
28. Пространства имен
29. Автозагрузка
30. Интерфейсы

9. Объектная модель

1. Класс-хинтинг
2. Передача объектов по ссылке

10. Исключения и практическое их применение

11. Ошибки в PHP, уровни ошибок

12. Этапы исполнения программы на PHP

В PHP есть 3 этапа:

Раннее связывание self – парсер, а позднее - runtime

1. Парсер – то что превращает текст программы в синтаксическое дерево, именно парсер проверяет корректность текста программы.
2. Кампилятор – превращает синтаксическое дерево в скомпилированную программу (байт код)
3. Runtime – когда скомпилированный код выполняется на виртуальной машине.

13. Чего нет в предыдущий двух версиях PHP по сравнению с актуальной?