

1. Как правильно настроить форму для отправки данных в обработчик? Какие атрибуты прописать, какие значения?

Action, method, enctype (не обязательный, если не грузим файлы).

В action пишем название файла, в котором будет обрабатываться форма.

В method пишем тип запроса, обычно post или get

В enctype multipart/form-data, что означает, что названия файлов не будут кодироваться и преобразоваться.

2. Что такое обработчик? Какую задачу он решает? А зачем выделять его в отдельный файл? В чем польза такого подхода?

Обработчик - файл со скриптом, который обрабатывает полученные данные. Выделяем в отдельный файл, что можно было многократно использовать код.

Например, загрузка файла на сервер может пригодиться на разных страницах сайта, а обработчик у них у всех будет один.

3. Какие методы бывают отправки данных через форму? В чем их разница? В каком случае использовать один, а когда другой?

Get, post.

Данные при get запросе передаются через браузерную строку, данные ограничены в длине, могут кешироваться и сохраняться браузером, небезопасный метод запроса. Используется, чтобы получить данные с сервера.

При post запросе данные скрыты и перехватить уже сложнее, могут быть любой длины, не кешируются и не сохраняются браузером. Используется, чтобы отправить данные на сервер, чаще всего из форм.

4. Опишите процесс записи строки в базу данных: как подключиться, подготовить запрос и выполнить

Для подключения к БД в последних версиях php используется класс PDO. Для подключения создаем новый экземпляр:

```
$db = new PDO ($dns, имя_пользователя, пароль, $options), где
```

```
$dns - "mysql:host=наш_хост;dbname=имя_бд"
```

```
$options – массив с параметрами, например можно указать,
```

```
[PDO::ATTR_DEFAULT_FETCH_MODE=>PDO::FETCH_ASSOC], чтобы каждый раз не прописывать этот атрибут в запросах
```

Для записи данных подготавливаем запрос:

```
$query = "insert into test_table(name) values(:name)";
```

```
Подготавливаем запрос: $stmt = db->prepare($query);
```

```
Отправляем $stmt->execute(["name" => "John"]);
```

5. Какой глобальный массив используется при создании флеш-сообщений?

```
$_SESSIONS
```

6. Опишите пошагово как создать и вывести флеш-сообщение

Открываем сессию: `start_session();`

Записываем в `$_SESSIONS` наше сообщение: `$_SESSIONS["message"] = "hello";`

Готово, выводить через `echo $_SESSIONS["message"];`

7. Как проверить текст на формат эл. адреса средствами php?

С помощью функции `filter_var($email, FILTER_VALIDATE_EMAIL)`

8. Какие функции используются для генерации пароля и его проверку?

Используется функция хеширования: `password_hash($pwd, PASSWORD_DEFAULT);`

Для проверки: `password_verify($pwd, $hash);`

9. Почему важно сохранять пароль в виде хэша? Почему не рекомендуется хранить пароль в открытом виде?

Когда мы храним данные в виде хэша, их становится невозможно использовать в целях взлома.

10. Напишите какая есть особенность у веб-сервера которую покрывает Сессия? Иными словами, зачем нужна Сессия?

Этот глобальный массив нужен для сохранения различной информации пользователя на все время его пребывания на сайте (или даже дольше, если мы настроили время жизни сессии в настройках сервера)

11. Опишите пошагово реализацию функционала Аутентификации(входа в систему).

Создаем форму с именем пользователя и паролем. По кнопке передаем данные обработчику, который проверяет их валидность, а затем проверяет есть ли такой пользователь в бд. Если пользователь есть и данные валидны, то мы авторизуем пользователя, не забыв проверить правильность пароля. Если пользователя с таким именем не существуем – вносим новую запись в бд.

12. Какая есть особенность настройки формы для загрузки файлов?

Необходимо указать атрибут `enctype = "multipart/form-data"`

13. Какое значение должно быть у инпута для множественной загрузки файлов?

Необходимо добавить `[]` в названии атрибута `name` и добавить атрибут `multiple`

14. Как создать уникальное название файла?

Использовать встроенную функцию `uniqid()`

15. Напишите названия функций для загрузки файла, проверки существует ли файл в папке, удаления файла

```
Move_uploaded_file($from, $to);  
File_exists($file);  
Unlink($file);
```

16. Какой глобальный массив хранит все файлы передаваемые через форму?

`$_FILE`

17. Опишите структуру глобального массива хранящего файлы: какие имеет ключи и значения

```
[name] => имя  
[type] => mime тип файла  
[tmp_name] => временный путь до загруженного файла  
[error] => ошибка  
[size] => размер файла
```

18. Помогите человеку решить задачу. «Пользователь загрузил файл и нажимает отправить. Он ожидает, что файл сохранится в папку, а название в базу. Файл сохраняется в папку, но в базу название не сохраняется» В чем может быть ошибка?

Ошибка может быть в подключении к бд, неправильно выбранной таблице, запросе, ошибке в переданной переменной при отправке запроса

19. Помогите человеку решить задачу. «Создал массив элементов, и хочу вывести при помощи цикла. Вроде бы все написал, но выходит ошибка: illegal offset type» Что нужно сделать, чтобы решить эту проблему?

Скорее всего ошибка в ключе массива, он должен быть или строкой или числом.

20. Что означает эта ошибка. «Parse error Unexpected (, expected ;» Что нужно сделать, чтобы избавиться от ошибки?

Интерпретатор ожидает точку с запятой, скорее всего забыли закрыть скобку.

21. Помогите человеку решить задачу. «Создал форму с инпутом. Пишу текст и отправляю. В обработчике `$_POST` пустой» В чем может быть ошибка?

Скорее всего не указан атрибут name или указан неправильно

22. Чем технически отличается глобальный массив от обычного массива?

Глобальный массив предопределен заранее и связан логикой с другими встроенными функциями

23. Дайте человеку совет. Он говорит: «Когда я пишу код, я не обращаю внимания на названия переменных, функций и массивов. Для меня все и так понятно. Поэтому я даю короткие названия, типа \$a, \$b, \$smth. Но потом, понимаю, что мне тяжело ориентироваться в них»

Если человек, говорит, что «понимает», он уже на правильном пути. Названия переменных не должны быть просто контейнерами, а должны прежде всего отражать суть и свое назначение, чтобы любой, кто наследует твой код, мог легко его поддерживать

24. Дайте человеку совет. Он говорит: «Хочу самостоятельно написать регистрацию и авторизацию. Приступаю и у меня ступор: не знаю с чего начать. Вроде бы понимаю, что регистрация — это добавление пользователя в базу, но не понимаю с чего начинать?»

Я бы посоветовал сложную задачу разбить на маленькие. Какие данные нам будут нужны? Нужно ли проверять данные, как обеспечить уникальность и безопасность, затем разбить все это на этапы.

Но вообще если человек в ступоре от таких простых алгоритмов, то он или не пытался думать сам или не может думать в принципе.

25. Объясните человеку в чем разница между \$\_SESSION и \$\_COOKIE: «Не пойму в чем разница между сессией и куки? Вроде бы одно и то же...пока полностью не осознал.»

Куки открытая информация, которую легко украсть, так как хранится в открытом виде в браузере пользователя и передается с каждым запросом.

Сессия хранится на сервере и используется только при необходимости

Куки пользователь может удалить, в то время как время жизни сессии и данных, которые она хранит определяет разработчик.

26. Дайте человеку совет. Он говорит: «Скачал кучу курсов. По одним названиям можно понять, что я скоро точно стану программистом. Но сколько бы я не смотрел, у меня не получается самостоятельно написать даже простенький скрипт. Что я делаю не так? Как надо подходить к обучению?»

Надо остановиться на каком-то одном курсе и постараться дойти до конца

27. Какие у вас впечатления от первой части практики?

Теорию практически не смотрел, лишь иногда обращался по мере решения заданий. Сами задания довольно неплохие, хотя в самом начале немного смутил большой блок с массивами, поэтому плюнул и уже на пятом задании перешел на реализацию через бд, показалось так веселее, да и хотелось чтобы руки привыкли писать в консоле sql код) Структура понравилась, не устал, было интересно. Вопросы вот на 10 и этом занятии напрягают ленивых личностей вроде меня, это да, очень много времени убивают, но уверен есть польза в том, чтобы сесть и еще раз все вспомнить и обдумать, написать ответы.

Хотелось бы чтобы в заданиях не было подсказок или они были скрыты под спойлером. Тупо тз и в бой. Вот вам спойлеры, вот вам видео на всякий случай, этого было бы достаточно думаю. А то на некоторых заданиях там в тексте уже решение полноценное со всеми шагами и функциями)