

## ПРОБЛЕМА ПЕРЕДАЧИ ЗАГОЛОВКОМ МЕЖДУ СТРАНИЦАМИ В PHP

Общение браузера с сервером происходит по протоколу HTTP. Когда вы набираете адрес, или нажимаете на ссылку, браузер посылает HTTP запрос серверу. Сервер отвечает. Первыми в ответе всегда идут HTTP заголовки. И только потом сервер посылает, а браузер принимает данные [1]. Из-за этого правила – сначала заголовок, а потом информация, и происходит эта ошибка. PHP посылает заголовки автоматически, как только скрипт начинает выдавать браузеру информацию. Если хоть один символ был уже передан пользователю, заголовки уже ушли, и снова их отправить невозможно, а команды `header()`, `setcookie`, `session_start()`, посылают HTTP заголовки.

Для решения этой проблемы нужно функцию `header()` (или `setcookie`, `session_start()`) и логику, которая ее вызывает, поместить ДО любого вывода в браузер. Планируйте структуру скрипта: блок, который обрабатывает POST, не должен ничего выводить в браузер.

**Буферизация вывода (БВ)** является обходным путем для решения проблемы. Она часто работает надежно, но не может заменить надлежащее структурирование приложения и отделение вывода от управляющей логики. Его цель – свести к минимуму разрозненные передачи на веб-сервер. Настройка `output_buffering=` может помочь. Настройте БВ в `php.ini` или через `.htaccess` или даже `user.ini` в установках FPM/FastCGI. Включение этого параметра позволит PHP БВ перед передачей на веб-сервер. Его также можно задействовать вызовом `ob_start()`; поверх сценария вызова. Что, однако, менее надежно по нескольким причинам: если `<?php ob_start(); ?>` запускает первый сценарий, пробелы или спецификация могут быть перетасованы раньше, что сделает его неэффективным. Он может скрывать пробелы для вывода HTML. Но как только логика приложения пытается отправить двоичный контент (например, сгенерированное изображение), БВ вызовет проблемы. Буфер ограничен по размеру и может переполниться. Поэтому оба подхода могут стать ненадежными, а значит БВ считается просто костылем.

### ЛИТЕРАТУРА

1. Шаблоны [Электронный ресурс] / Сайт PHPFAQ – 2021. – Режим доступа: <http://phpfaq.ru/tech/tpl>– Дата доступа: 01.04.2022.