**PHP**

***Шайдуллинов Расул Набиевич***

***Букенова Индира Нурмухамбетовна***

***Төлеушова Айнур Төлендіқызы***

*Студент, Алматинский Технологический Университет,*

*Казахстан, г. Алмата*

**PHP**

***Shaidullinov Rasul Nabievich***

***Bukenova Indira Nurmukhambetovna***

***Toleushova Ainur Tölendikyzy***

*Student, Almaty Technological University,*

*Kazakhstan, Almaty*

**АННОТАЦИЯ**

В этой статье рассмотрен скриптовый язык – PHP. Сама статья подойдет для людей не шибко сведущих в программировании. Тут вы найдете ответы на вопросы по типу: “Что такое PHP?”, “Как оно появилось?”, “ Где применяется?” и “Хорош ли язык?”.

**ABSTRACT**

This article deals with the scripting language PHP. The article itself is suitable for people who are not very knowledgeable in programming. Here you will find answers to questions such as: “What is PHP?”, “How did it appear?”, “Where is it used?” and “Is it a good language?”.

**Ключевые слова:** PHP, история, применение, преимущества, недостатки

**Keywords:** PHP, history, application, advantages, disadvantages

PHP-один из самых распространенных языков web-разработки. На нём делают сайты и веб-приложения — блоги, интернет-магазины или социальные сети. На сегодняшний день, PHP используют более чем 80% сайтов в интернете.

**Что такое PHP?**

PHP – это скриптовый язык общего назначения, интенсивно применяемый для разработки веб-приложений, исполняющихся на Web-сервере. Аббревиатура PHP означает “Hypertext Preprocessor (Препроцессор Гипертекста)". Синтаксис языка берет начало из C, Java и Perl. PHP достаточно прост для изучения. Преимуществом PHP является предоставление web-разработчикам возможности быстрого создания динамически генерируемых web-страниц.

**История развития PHP**

Истоки PHP лежат в старом продукте, имевшем название PHP/FI. PHP/FI был создан Расмусом Лердорфом в 1995 году и представлял собой набор Perl-скриптов для ведения статистики посещений его резюме. Первоначально он использовал их для отслеживания посещений своего веб-резюме и назвал этот набор скриптов "Personal Homepages Tools". Со временем требовалось все больше функциональных улучшений, и Расмус переписал PHP Tools, создав более крупную и богатую реализацию. Эта новая реализация могла взаимодействовать с базами данных и многое другое, что создало основу, с помощью которой пользователи могли создавать простые динамические веб-приложения. PHP/FI включал в себя базовую функциональность современного PHP. В нем были переменные в стиле Perl, автоматическая интерпретация форм и возможность встраивания в html-код. В 1997 году была выпущена версия PHP/FI 2.0. Вторая версия C-реализации определяла группу пользователей: несколько тысяч человек по всему миру, имевших около 50 000 доменов, что составляло примерно 1% от общего числа доменов Интернета.

***PHP3***

PHP 3.0 был первой версией, напоминающей PHP в том виде, в котором мы знаем его сегодня. В 1997 году Энди Гутманс и Зив Сураски переписали код с самого начала. Для совместной работы над PHP 3.0 с помощью базы разработчиков PHP/FI 2.0 Энди, Расмус и Зив решили объединиться и объявить PHP 3.0 официальным преемником PHP/FI. Одной из сильных сторон PHP 3.0 была возможность расширения ядра. Помимо предоставления пользователям надежной инфраструктуры из множества различных баз данных, протоколов и API, расширяемость PHP 3.0 привлекла многих сторонних разработчиков, желающих добавить в язык свои собственные модули. Важным шагом стало поддержка синтаксиса ООП и гораздо более мощный и последовательный синтаксис самого языка.

***PHP4***

К зиме 1998 года, почти сразу после официального выхода PHP 3.0, Энди Гутманс и Зеев Сураски начали перерабатывать ядро PHP. Новый движок 'Zend Engine' успешно справился с поставленными задачами разработчиков. PHP 4.0, основанный на этом движке и принесший с собой набор дополнительных возможностей, был официально выпущен в мае 2000 года. Помимо значительного повышения производительности, новая версия имела ряд других ключевых нововведений, таких как поддержка гораздо большего числа веб-серверов, поддержка HTTP-сессий, буферизация вывода, более безопасные способы обработки пользовательского ввода, а также несколько новых языковых конструкций.

***PHP5***

В PHP5 используется новая версия "движка" Zend - Zend Engine 2. В PHP5 объектная модель была значительно переработана. В то же время было добавлено много новых возможностей, благодаря которым PHP5 получил некоторые черты объектно-ориентированных языков, таких как C++ и Java. Изменения коснулись производительности встроенных стандартных функций PHP. Появились новые директивы конфигурационного файла php.ini. Добавлены новые функции, а также функции графической библиотеки GD. Поддержка XML в версии PHP5 стала полной, поддерживаются новые расширения DOM и XML.

**Где используется PHP?**

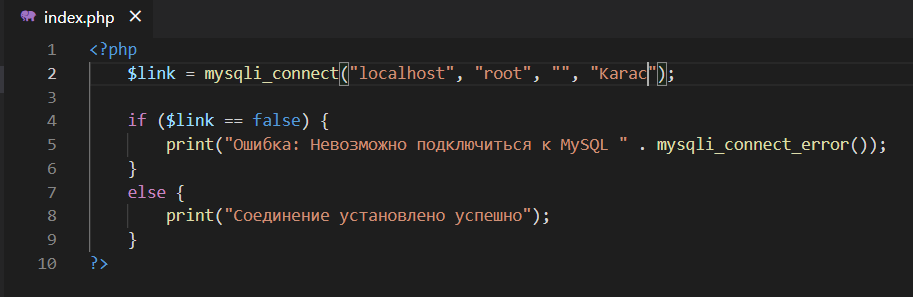
Область применения PHP сосредоточена на написании скриптов на стороне сервера. Он может обрабатывать данные форм, генерировать динамические страницы, отправлять и принимать cookies. Но PHP способен выполнять и многие другие задачи.

Существуют три основных области, где используется PHP:

1. Создание скриптов для выполнения на стороне сервера. Наиболее широко для этого используется PHP. Все, что вам нужно, это парсер PHP, веб-сервер (XAMPP/OpenServer) и браузер. Для того чтобы просмотреть результаты выполнения PHP-скриптов в браузере, вам необходим работающий веб-сервер и установленный PHP.
2. Создание скриптов для выполнения в командной строке. Вы можете создать PHP-скрипт, который будет выполняться независимо от веб-сервера и браузера. Все, что вам нужно - это парсер PHP. Этот способ использования PHP идеально подходит для скриптов, которые должны выполняться регулярно, например, с помощью cron(Linux) или планировщика задач на платформах Windows.
3. Создание GUI-приложений, работающих на стороне клиента. Возможно, PHP не самый лучший язык для создания таких приложений, но если вы хорошо знаете PHP и хотите использовать некоторые его возможности в своих клиентских приложениях, вы можете использовать PHP-GTK для создания таких приложений.

**Базы данных**

К сайту PHP подключается база данных. Это во многом упрощает разработку, потому что вся информация будет храниться в БД, а не в самом коде. Так быстрее редактировать и добавлять данные на сайт, не меняя сам сайт. В БД хранятся данные о пользователях (имя, фамилия, логин, пароль и т.д.), книгах, сериалах, играх и т.п.. Если б не было БД, все данные хранились бы в HTML-коде.

**Пример подключения к БД**

*Где: "localhost" – адрес сервера;*

*"root" - имя пользователя;*

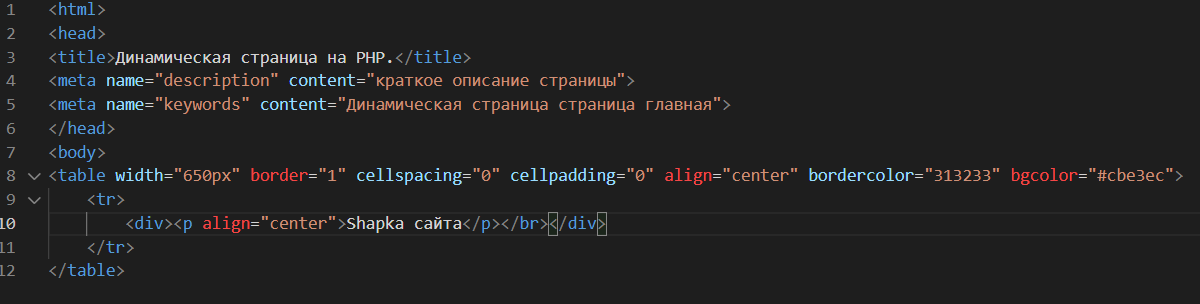
*"" – пароль;*

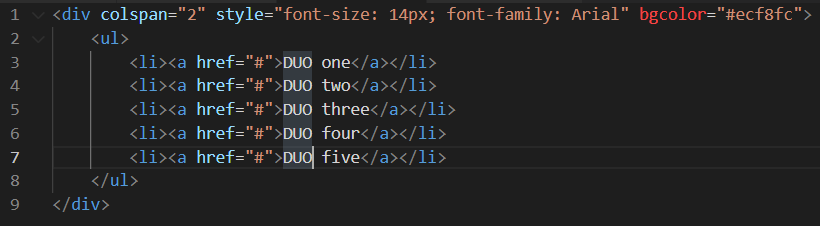
*"Karac" - имя базы данных.*

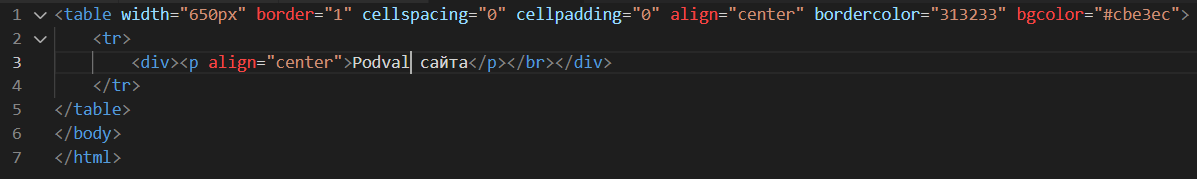
Так как мы можем менять данные на сайте, не меняя сам сайт, то PHP хорошо подходит для создания CMS(Content Management System), интернет-магазинов и всевозможных интернет-сервисов.

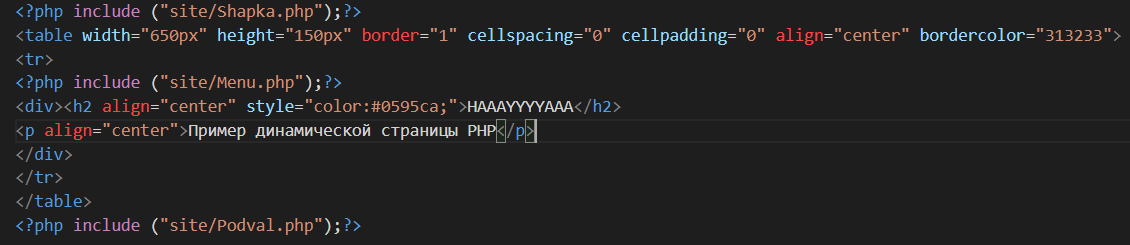
**Динамичные страницы**

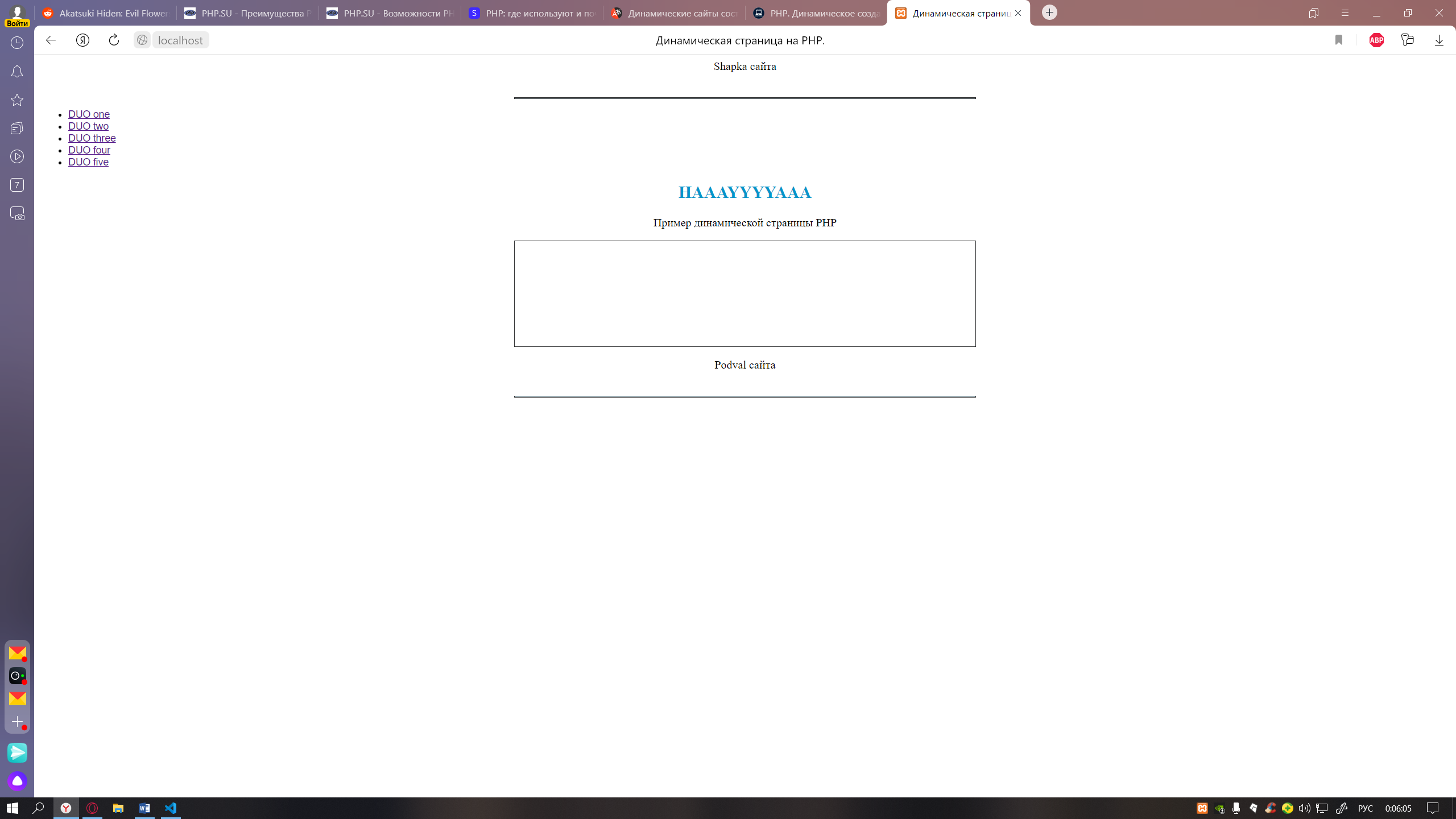
Динамические сайты состоят из динамических веб-страниц, которые могут реагировать на действия пользователя и изменяться. Такие страницы генерируются веб-сервером из нескольких файлов (шаблонов). Вся информация хранится в базе данных. Когда пользователь запрашивает определенную страницу, нужная информация извлекается из БД и вставляется в шаблон сайта, образуя web-страницу, и отправляется обратно пользователю. В следствии, чтобы обновить информацию на сайте, нужно просто добавить текст для новой страницы в БД. Это экономит массу времени и места, потому что не нужно каждый раз прописывать этот код заново.











**Пример создания динамических страниц на PHP**

**Преимущества и Недостатки**

Преимущества:

1. Традиционность – в связи тем, что многие конструкции языка были “позаимствованы” C и Perl, он снижает усилия программистов к его изучению;
2. Простота – вы можете начать код PHP c “<?php”, а заканчивается с “?>”;
3. Эффективность – из-за своего движка, PHP является транслирующим интерпретатором. По этой причине PHP обрабатывает сценарии с большой скоростью;
4. Гибкость - поскольку РНР является встраиваемым языком, он отличается исключительной гибкостью по отношению к потребностям разработчика;
5. Open Source – открытый код помог в развитии PHP.

Недостатки:

1. Смешанный код – тут приходится работать в файлах с HTML-версткой(можно обойти с помощью шаблонов);
2. Простота – из-за простоты PHP, на нем было написано больше 80% сайтов в интернете. Эта самая простота стала причиной появления большого числа сайтов с говно-кодом;
3. Безопасность – на PHP более низкая защищенность, чем на других языках
4. Малое количество фреймворков и пакетов.

**Заключение**

PHP - это язык, с помощью которого можно реализовать практически любую функциональность. Хотя многие разработчики его недолюбливают, он отлично справляется со своими задачами.

**Список литературы:**

1. <https://ru.wikipedia.org/wiki/PHP>
2. <https://www.php.net/>
3. <http://www.php.su/>