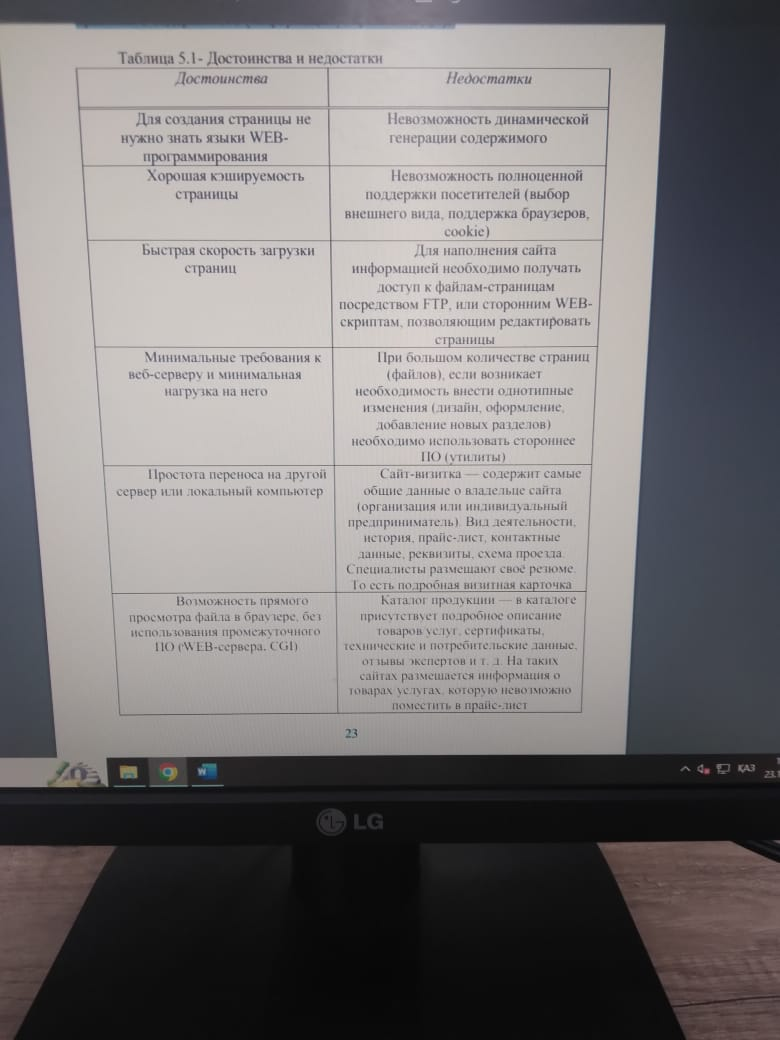
**Лекция №5. Статитические и динамические WEB-сайты. Технология** CGI.

Цель лекции: ознакомление со статитическими и динамическими страницами, а также одной из технологией, которая позволяет добавлять динамическое содержание в WEB-страницы.

**Содержание лекции:** достоинства и недостатки WEB-сайтов. Технология CGI.

Статический WEB-сайт — сайт, состоящий из статичных html (htm, dhtml, xhtml) страниц составляющих единое целое. Содержит в себе (в виде HTML-размеченных) текст, изображения, мультимедиа содержимое (аудио, видео) и HTML-теги. Теги бывают как служебные, предназначенные для обозревателя, так и предназначенные для размещения, формирования внешнего вида и отображения информации. Все изменения на сайт вносятся в исходный код документов (страниц) сайта, для чего необходимо иметь доступ к файлам на WEB-сервере. Обычно статический WEB-сайт представляет собой набор файлов HTML, размещенных на WEB-сервере. Достоинства и недостатки представлены в таблице 5.1. Динамический WEB-сайт — сайт, состоящий из динамичных страниц — шаблонов, контента, скриптов и прочего, в виде отдельных файлов. Страница сайта, показываемая в итоге браузеру пользователя, формируется на 23 стороне сервера динамически, по запросу, из страницы-шаблона и отдельно хранимого содержимого (информации, скриптов и др.).



Редактирование собственно содержимого (как и страницы-шаблона) может производиться как средствами самого сайта, так и с применением стороннего ПО. Возможность править все страницы предоставляется только определенной категории пользователей (например администраторам, или же зарегистрированным пользователям). Как правило для отображения любого количества однотипных страниц используется одна страница-шаблон, в которую подгружается соответствующее содержимое, это позволяет единомоментно корректировать внешний вид сайта (множество всех его страниц), редактируя всего лишь один шаблон. В некоторых случаях к правке определенного веб-контента допускаются анонимные пользователи, что бывает реже (например, на форумах — добавление сообщений). Процесс получения содержимого сайта обычно выглядит следующим образом: 1) Генерация содержимого на стороне сервера. 2) Передача сгенерированной странички клиенту. 3) Генерация содержимого на стороне клиента. Генерация содержимого на стороне сервера. Сервер получает запрос от Клиента (например, page.ru/index.php) и запускает обработку файла-скрипта (в данном случае - index.php) интерпретатором. Языки программирования на Серверной стороне используются разные, наиболее часто встречаются, например: Java, PHP, Perl, ASP.NET, Ruby и другие. Именно на этой стадии происходит выборка необходимой информации из баз данных и наполнение ею страницы, после чего готовая страница передаётся Клиенту. Генерация содержимого на стороне клиента. После того, как страница получена Клиентом с Сервера, программабраузер обрабатывает её и отображает Пользователю, при этом исполняя скрипты Клиентской стороны, если они были указаны в странице и получены. На Клиентской стороне используется JavaScript, который может использоваться как для минимальных, невидимых глазу операций (например, проверки на правильность задания паролей при регистрации на сайтах), так и глобальных последовательностей и приложений. Комбинированная генерация. Чаще всего в жизни встречается именно комбинация этих двух методов генерации — весь «новый интернет» основан на нём, это, и «умная строка с подсказкой вариантов» у поисковых систем, и «редакторы» в форумах, и всплывающие меню, и многое другое. После набора в адресной строке браузера необходимого URL, браузер получает информацию об используемом протоколе (HTTP) и имени сервера. Браузер обращается к DNS-серверу, для того чтобы перевести имя сервера в IP-адрес. На основе полученного IP-адреса браузер устанавливает связь с 25 искомым WEB-сервером и, используя протокол HTTP, запрашивает искомый ресурс. Сервер посылает браузеру HTML-страницу, которая хранится на сервере. Браузер считывает HTML-тэги, воссоздает страницу на экране вашего компьютера, и вы видите результат вашего запроса. Обычно даже простая WEB-страница содержит не только текст, но и графику, т. е. состоит из нескольких файлов разного типа. Для того чтобы браузер мог распознавать эти файлы, сервер информирует о том, какой файл будет отправляться (текст в формате HTML, или графика в формате JPG и т. д.), и после этого передается содержание файла. Несмотря на тот факт, что обычно WEB-страница состоит из многих файлов, за время одного запроса WEB-сервер посылает только один файл. То есть, получив HTML-текст и найдя в нем, например, ссылку на графический элемент, браузер посылает новый запрос на сервер (причем необязательно на тот же самый сервер, с которого пришел HTML-текст). Для того чтобы скачать каждый новый файл, браузер должен послать новый HTTP-запрос. При этом, поскольку современные серверы и браузеры работают в многопоточном режиме, одновременно может выполняться несколько запросов. Если запрашиваемого ресурса по данному адресу нет, то WEB-сервер выдаст сообщение 404/File not found (404/Файл не найден). Технология CGI. Страницы, которые были созданы по вашему запросу динамически - «сформированы на лету». Например, любая книга отзывов предоставляет вам определенную форму, в которую вы добавляете свой текст и затем, когда вы в следующий раз открываете эту страницу, она содержит новое сообщение. Одной из технологий, позволяющих добавлять динамическое содержание в WEB-страницы, является CGI(Common Gateway Interface). Она позволяет ассоциировать с тем или иным URL не статический документ, а программу, в результате работы которой могут быть сформированы данные в реальном времени. Например, если вы хотите предоставлять мгновенные погодные данные в некотором районе, вы должны каждый раз генерировать новую страницу. Это может быть реализовано на основе CGI-технологии. В момент обращения на сервер на нем запускается CGI-программа, которая обращается к цифровому измерительному устройству и выдает сведения о температуре, давлении и так далее. Каждый раз, обращаясь на этот адрес за справкой, вы будете получать текущие значения. Другой пример: когда вы заполняете некоторый запрос в поисковой системе, то в ответ получаете результат работы CGI-программы в виде набора искомых адресов. CGI-программу можно представить как часть WEB-сервера, которая запускается WEB-сервером в реальном времени. Сервер передает запросы пользователя CGI-программе, которая их обрабатывает и возвращает результат своей работы на экран пользователя. С точки зрения клиента нет никакой разницы, что находится по адресу конкретного URL - статический документ или CGI-программа. Браузер воспринимает информацию одинаково вне зависимости от того, сформирована ли она «на лету» или это статическая 26 страница. Результат работы CGI-программы имеет тот же формат, что и статический документ. Под термином CGI понимают не только программу, но и протокол. В этом смысле CGI - это стандартный способ для WEB-сервера передать пользовательский запрос прикладной программе и получить данные для передачи их пользователю. Протокол CGI пересылки данных между сервером и приложением является частью протокола HTTP. Большая часть CGI-программ является CGI-скриптами. Скриптом называют набор инструкций, которые интерпретируются или выполняются другой программой. Некоторые языки, такие как Perl, JavaScript, задумывались именно как скриптовые языки или, как их еще называют, языки описания сценариев. В принципе CGI-программы могут быть написаны как на скриптовых языках, так и на компилируемых: С, C++, Delphi. Альтернативой технологии CGI является технология компании Microsoft, называемая Active Server Page (ASP), которая построена по тому же принципу: скрипт, включенный в WEB-страницу, выполняется на сервере до того, как страница отсылается пользователю. Существует также ряд других технологий, работающих по тому же принципу. В отличие от схемы, где динамическое содержание формируется на стороне сервера, можно также генерировать динамическое содержание страниц уже на стороне клиента. В последнем случае активные документы хранятся на WEB-серверах и скачиваются на локальный компьютер так же как и статические страницы, а после того, как программа активного документа оказывается на локальном компьютере, она выполняет на нем определенные вычисления, и результаты этих вычислений отображаются на экране. Соответственно, скорость отображения данных на экране не зависит от скорости связи с удаленным сервером, поскольку вывод активного документа происходит только после его загрузки на локальный компьютер. Следует при этом отметить, что загрузка программы, служащей для формирования документа, может занимать весьма ощутимое время. Для создания активных документов могут использоваться различные технологии: это могут быть приложения, написанные на JavaScript, Java-апплеты и элементы управления ActiveX.