**Коды ответов сервера полный список**

В Вебе глобально существуют 2 роли - “Клиент” и “Сервер”. Они общаются друг с другом посредством языка - HTTP (протокол передачи гипертекста).

В роли транспорта при передаче запросов выступает TCP/IP - коммуникационный Протокол Управления Передачей/Интернет Протокол.

**Клиент** - это любое устройство (компьютер, мобильный телефон) подключенное к сети Интернет.

**Сервер** - это компьютер - место, где находятся все веб-страницы, файлы сайта или приложения. Это, своего рода, “дом” сайта.

**DNS-сервер** - это сервис, который связывает имя сайта (например, google.com) с его реальным адресом - “домом” сайта, сервером.

Итак, вы вводите название сайта в адресной строке браузера и хотите на него перейти. Что происходит дальше?

1. Сначала, браузер отправляет запрос DNS-серверу, чтобы выяснить, где живет сайт, а именно - где находится сервер сайта?
2. Затем, отправляется запрос к серверу (дому сайта).

И тут возможно множество вариантов ответа сервера. Давайте их разберем детальнее.

**Что такое код ответа сервера**

**Код ответа сервера** (код состояния HTTP, англ. HTTP status code) - это трехзначное число, состоящее из 3-х арабских цифр. За кодом ответа следует поясняющая фраза, которая объясняет причину ответа. 1 цифра кода - поясняет класс ответа.

**Классы кодов ответа сервера**

Ответы сервера делятся на 5 классов:

* **1хх - Информационный (Informational)** - означает, что сервер принял запрос и обрабатывает его дальше.
* **2хх - Успешно (Success)** - означает, что сервер успешно обработал запрос.
* **3хх - Перенаправление (Redirection)** - сервер сообщает, что для выполнения запроса необходимы дополнительные действия (например, переход на другой URL)
* **4хх - Ошибка клиента (Client Error)** - сервер не может успешно выполнить запрос по причине ошибки на стороне клиента.
* **5хх - Ошибка сервера (Server Error)** - сервер не может успешно выполнить запрос по причине ошибки на стороне сервера. Рядом отображается детальное объяснение причины ошибки.

**1 класс ответов сервера - “Informational”**

1 класс ответов сервера - Informational

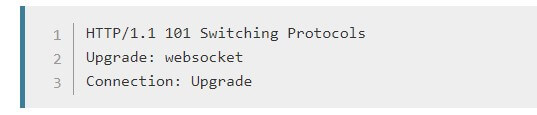
**100 Continue (Продолжить)**

Это промежуточный ответ сервера, информирующий, что сервер принял запрос и будет дальше совершать обработку.

**101 Switching Protocols (Переключение протоколов)**

Этот код присылается в ответ на запрос клиента, содержащий заголовок Upgrade. Он означает, что сервер переключился на указанный в заголовке протокол. Сервер отправляет заголовок ответа Upgrade, указывая протокол, на который он переключился.

Пример



Пример 101 кода ответа сервера

**102 Processing (В обработке)**

Сервер получил запрос, обрабатывает его, но обработка ещё не завершена и требует дополнительного времени. Клиент не должен разрывать соединения, а обнулить таймер и дождаться следующего ответа. Данный код используется в протоколе WebDAV (надстройка HTTP для работы с файлами).

**103 Early Hints (Ранние подсказки)**

Прежде всего, данный код используется с заголовком Link. Используется чтобы клиент мог получить предварительные для загрузки ресурсы, пока сервер готовит основной ответ.

**2 класс ответов сервера - “Success”**

2 класс ответов сервера - Success

**200 OK (Успешно)**

Самый популярный ответ сервера - означает успешную обработку запроса.

Успешно получены данные в теле или в заголовке письма. Например, страница была успешно загружена.

Запрос успешно обработан. Что значит "успешно", зависит от метода HTTP, который был запрошен:

* GET: "ПОЛУЧИТЬ". Запрошенный ресурс был найден и передан в теле ответа.
* HEAD: "ЗАГОЛОВОК". Заголовки переданы в ответе.
* POST: "ПОСЫЛКА". Ресурс, описывающий результат действия сервера на запрос, передан в теле ответа.
* TRACE: "ОТСЛЕЖИВАТЬ". Тело ответа содержит тело запроса полученного сервером.

**201 Created (Создано)**

Код обозначает успешное создание нового ресурса, в результате запроса клиента. Новый ресурс обозначается в виде URL-адреса в теле письма либо в заголовке Location.

Этот код обычно присылается в ответ на запрос PUT "ПОМЕСТИТЬ".

**202 Accepted (Принято)**

Уведомляет клиента, что запрос принят, но его обработка может занять длительное время. Ожидать завершения запроса не обязательно. Может быть предназначен для пакетной обработки, либо когда запрос обрабатывается другим сервером/процессом.

**203 Non-Authoritative Information (Неавторитетная информация)**

Код сообщает об успешной обработке запроса, однако предупреждает, что информация предоставлена не исходным сервером, а другим источником (резервная копия, другой сервер) и поэтому актуальность не подтверждена.

**204 No Content (Нет содержимого)**

Ответ не содержит тела сообщения, а только заголовки (метаданные), которые могут быть полезны. Может используется при запуске скриптов, когда не нужно перезагружать и менять страницу. Клиент может использовать их для обновления кешированных заголовков полученных ранее для этого ресурса. Также, например, может выполняться, когда пользователь кликает по пустому месту страницы или по изображению без ссылки.

**205 Reset Content (Сбросить содержимое)**

Сервер сообщает браузеру очистить формы ввода данных (сбросить заполненные формы, поля и т.д.). Обновление при этом не требуется, тело сообщения сервера пустое.

**206 Partial Reset (Частичное содержимое)**

Код обычно относится к процессу кеширования. Клиент запрашивает от сервера только определенный диапазон (часть данных) - заголовок Content-Range.

**207 Multi-Status (Мультистатусный)**

Сервер успешно обработал несколько независимых операций. Результаты помещаются в теле сообщения: XML-документ с объектом Multistatus.

**226 IM Used (Использовано IM)**

Успешное завершение операции по запросу A-IM и возврат содержимого с учетом указанных параметров.

**3 класс ответов сервера - “Redirection”**

3 класс ответов сервера - Redirection

**300 Multiple Choices (Множественный выбор)**

Указанный URL предлагается в нескольких вариантах. Например по MIME-типу (тип файла - аудио, текстовый, графический и др.), языковому признаку и другим свойствам. В сообщении сервера передаются все возможные альтернативы ресурса с возможностью выбора нужного варианта.

**301 Moved Permanently (Перемещение на постоянной основе)**

Указанный URL был изменен или не существует и пользователю предлагается переход на новый URL. Код ответа 301 означает, что адрес перемещён окончательно (навсегда). В теле сообщения указывается новый URL в поле Location.

Для поисковых систем 301 редирект - это сигнал, что необходимо “склеить” старый проиндексированный URL с новым и добавить его в индекс.

**302 Found (Найдено)**

Указанный URL был временно перенесен на другой адрес. Новое расположение указывается в поле Location.

Новые  изменения в  URL  могут быть  доступны в  будущем. Таким образом, этот URL, должен быть использован клиентом в будущих запросах.

**303 See Other (Смотреть другое)**

URL был изменен и клиенту необходимо перейти по новому URL, указанному в поле Location c использованием метода GET.

Для того, чтобы старый URL оставался в индексе необходимо, чтобы он отвечал кодом 200.

**304 Not Modified (Не изменено)**

Данный код используется при кэшировании и означает, что ресурс не был изменён. Клиент может продолжать использовать версию из кеша. Робот может индексировать страницу, не загружая её. Нагрузка на сервер не увеличивается.

**305 Use Proxy (Использовать прокси)**

Код ответа сообщает, что доступ к ресурсу возможен только через прокси-сервер, указанный в поле Location заголовка. Данный код могут отдавать только родные HTTP-сервера (не прокси). Этот код связан с безопасностью данных.

**306 Reserved (Зарезервировано)**

В настоящее время код не используется и зарезервирован. Использовался в ранних спецификациях. Изначально подразумевалось, что " последующие запросы должны использовать указанный прокси."

**307 Temporary Redirect (Временное перенаправление)**

Код 307 похож на код 302 (временное перемещение) и говорит нам, что адрес ресурса временно изменен на новый, но позже вернется обратно.

307 код отличается только тем, что метод запроса (POST/GET) не изменяется.

Код 307 по спецификации более новый, чем 302.

**308 Permanent Redirect**

Код 308 - Это означает, что ресурс теперь постоянно находится в другом URI, указанном в заголовке Location: HTTP Response. Аналог 301, с той разницей, что не позволяет менять метод запроса (POST/GET). Код 308 передает вес на новую страницу, так же, как и 301.

Код 308, например, может быть использован вместо 301 на сайте с большим количеством форм, работающих методом POST.

**4 класс ответов сервера - “Client Error”**

4 класс ответов сервера - Client Error

**400 Bad Request (Плохой запрос)**

Запрос не распознан сервером по причине синтаксической ошибки. Предлагается исправить синтаксис запроса. Помогает изменение запроса, а иногда - простая очистка кеша.

**401 Unauthorized (Неавторизован)**

Для получения запрашиваемого ответа нужна аутентификация. Статус похож на статус 403, но в этом случае, аутентификация возможна.

**402 Payment Required (Требуется оплата)**

Код в данный момент не используется и зарезервирован на будущее для цифровых платежных систем.

**403 Forbidden (Доступ запрещен)**

Запрос получен, но сервер не может его выполнить по причине ограничений доступа. У клиента нет прав доступа к содержимому, поэтому сервер отказывается дать надлежащий ответ.

**404 Not Found (Не найдено)**

Сервер получил запрос, но не нашел документа по запрашиваемому URL. Наиболее частая причина - файл находится по другому адресу, ссылку указали с ошибкой либо страница была удалена (в данном случае - более корректен код 410).

Код 404 - наиболее распространен в Вебе, как и код 200.

**405 Method Not Allowed (Метод не поддерживается)**

Код 405 означает, что сервер не применяет указанный метод к данному ресурсу. Сервер знает о запрашиваемом методе, но он был деактивирован и не может быть использован. Два обязательных метода,  GET и HEAD,  никогда не должны быть деактивированы и не должны возвращать этот код ошибки.

**406 Not Acceptable (Не приемлемый запрос)**

Обозначает, что сервер понял и принял запрос, но указанные параметры документа (формат/язык/кодировка..) не позволяют выполнить запрос.

**407 Proxy Authentication Required (Требуется авторизация прокси)**

Этот код ответа аналогичен коду 401, только аутентификация требуется для прокси сервера.

**408 Request Timeout (Время ожидания истекло)**

Код обозначает, что сервер оборвал неиспользуемое соединение, так как не получил ни одного запроса от пользователя в определенный промежуток времени. Код 408 не возвращается, если соединение прервано по инициативе клиента.

**409 Conflict (Конфликт)**

Сервер не может выполнить запрос из-за конфликтного обращения к ресурсу. Может быть использовано при совместной работе с файлами. Этот ответ отсылается, когда запрос конфликтует с текущим состоянием сервера.

**410 Gone (Удалён)**

В данном ответе сервер сообщает, что документ ранее находился на запрашиваемом адресе, но сейчас удалён навсегда. При этом, серверу не известно новое местонахождение документа. Если документ может быть восстановлен, то корректнее отдать код 404 и робот вернется проверить страницу в будущем.

**411 Length Required (Требуется указание длины)**

Сервер требует указать заголовок Content-Length, а иначе не может выполнить запрос.

**412 Precondition Failed (Условие ложно)**

Сервер не может успешно обработать запрос. Необходимо выполнить все условия для полей заголовков клиента (несоответствие при проверке одного или более условий).

**413 Request Entity Too Large (Размер запроса слишком большой)**

Сервер не может выполнить запрос, так как объем запрошенной информации превышает лимит. Если это временная проблема - указывается время ожидания в поле Retry-After.

**414 Request-URL Too Long (Слишком длинный запрашиваемый URL)**

Сервер не может обработать запрос, так как указан слишком длинный URL. Это довольно редкая ошибка. Например, этот ответ может встретиться когда клиент указывает длинные параметры через метод GET, а не POST.

**415 Unsupported Media Type (Неподдерживаемый тип данных)**

Сервер не может выполнить запрос, так как клиент использует неподдерживаемый формат данных.

**416 Requested Range Not Satisfiable (Недопустимый запрашиваемый диапазон)**

Сервер не может выполнить запрос из-за расхождений в указанных и фактических размерах ресурса. Клиент указал в поле заголовка Range значение, которое превышает фактический размер запрашиваемого ресурса. Например, при попытке получить часть файла, которая находится за пределами конца файла.

**417 Expectation Failed (“Ожидаемое” провалено)**

Данные из поля Expect не удовлетворяют требованиям сервера, поэтому запрос не может быть выполнен.

**422 Unprocessable Entity (Необработанный объект)**

Сервер поддерживает указанный тип данных, запрос имеет верный синтаксис, однако существует некая логическая/семантическая ошибка из-за которой невозможно выполнить запрос.

**423 Locked (Блокировка)**

Указанный ресурс недоступен из-за невозможности применения к нему данного метода.

**424 Failed Dependency (Ошибка зависимости)**

Сервер не может выполнить запрос, так как не выполнилась другая операция, от которой зависел данный запрос.

**426 Upgrade Required (Требуется обновление)**

От клиента требуется обновить протокол. Сервер указывает детали в полях Upgrade и Connection.

**5 класс ответов сервера - “Server Error”**

5 класс ответов сервера - Server Error

**500 Internal Server Error (Внутренняя ошибка сервера)**

Сервер не может выполнить запрос из-за некой внутренней ошибки. Сервер столкнулся с ситуацией, которую он не знает как обработать.  Например, нарушение конфигурации сервера, ошибки в файле .htaccess, ошибки пакета PHP или любые причины, не подпадающие под коды других ошибок класса 5xx.

**501 Not Implemented (Не может быть выполнено)**

Сервер не имеет возможности выполнить запрос. Чаще всего, сервер не понимает указанный в запросе метод. Единственные методы, которые сервера должны поддерживать (и, соответственно, не должны возвращать этот код) -  GET и HEAD.

**502 Bad Gateway (Неверный шлюз)**

Ситуация, когда запрос был отправлен прокси-серверу или серверу, выполняющему роль шлюза. В данном случае, “промежуточный” сервер запросил информацию у вышестоящего сервера и получил от него некорректный ответ.

**503 Service Unavailable (Сервис недоступен)**

Сервер не может выполнить запрос по техническим причинам. Чаще всего - перегрузка, обслуживание и др. Сервер может указать время, спустя которое можно повторить запрос в поле Retry-After.

**504 Gateway Timeout (Время ожидания шлюза истекло)**

Сервер, который выполняет роль шлюза не дождался ответа от вышестоящего сервера.

**505: HTTP Version Not Supported (версия HTTP не поддерживается)**

Сервер не может выполнить запрос из-за невозможности использования HTTP-версии, которая указана у клиента. В ответе сервер должен сообщить почему эта версия не поддерживается и какие другие протоколы поддерживаются этим сервером.

**506 Variant Also Negotiates (Experimental)**

Экспериментальный код. Указывает на ошибочное связывание, в результате внутренней ошибки конфигурации. Происходит зацикливание обращения сервера.

**507 Insufficient Storage (Недостаточно места)**

Сервер сообщает, что закончилось место и невозможно выполнить операцию. Это бывает при перезагрузке памяти сервера. Проблема может быть временной.

**510 Not Extended (Нет расширения)**

На сервере не доступно расширение, которое хочет использовать клиент для своего запроса. В ответе сервера должны быть указаны доступные расширения.

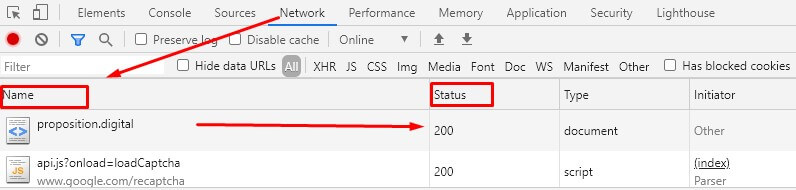
**511: Network Authentication Required (Необходима авторизация в сети)**

Клиенту сообщается, что необходимо авторизоваться для получения доступа к сети. В ответе возможно указания ссылки на ресурс для авторизации. Данный ответ исходит не от запрашиваемого сервера, а от сервера-посредника (например, провайдера).

**Как проверить ответ сервера**

Проверить ответ сервера вы можете прямо в браузере либо с помощью сервисов:

* Проверка в реальном времени в браузере. Необходимо открыть вкладку панели разработчика - нажать клавишу F12. Перейти во вкладку Network и обновить страницу. Вы увидите в столбце Name - адрес страницы, а в столбце Status - код ответа сервера.

Проверка кода ответа в браузере

* Онлайн-сервисы. Также можно воспользоваться онлайн-сервисами - bertal.ru, pr-cy.ru, checkmy.ru, sitechecker.pro и другими. Возможна как постраничная проверка, так и массовая (в зависимости от сервиса).
* Массовая проверка десктопными программами. Кроме того, можно массово протестировать все страницы сайта и получить коды ответов сервера с помощью десктопных программ-анализаторов сайтов (например, Screaming Frog).

Коды ответов сервера важны для поисковых систем и веб-мастеров. С их помощью можно управлять индексацией, распознавать и исправлять ошибки.