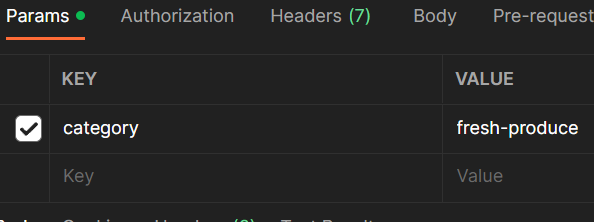
**Query** **Parameters**

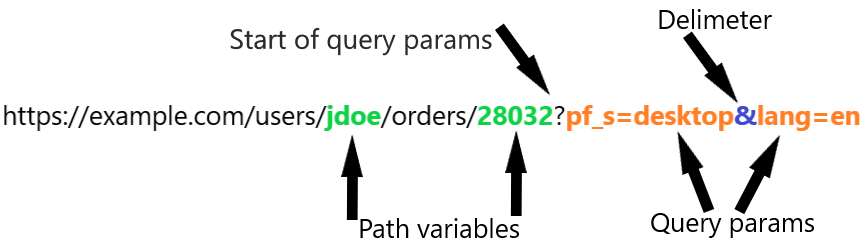


It is nothing but key-value pairs that we send along with the URL.

Parameters - с параметрами всегда работают с документацией, заранее нельзя предусмотреть какие параметры валидные (доступные), а какие нет, это полностью зависит от разработчика API, также как и с значениями того или иного параметра. В Postman-е для этого существует специальный UI с вкладкой Params, где можно редактировать параметры запроса и делать их активными или на время отключать при выполнении запроса, здесь же редактируются и path variables.

Основное назначение **Query** **Parameters** это фильтрация получаемых данных.

It is very useful in case the response is huge, and we dont need all the information from response, only specific.

в Postman-e ниже ‘KEY’ ввести название параметра, в поле VALUE - соответственно значение, а галочкой слева можно активировать/деактивировать участие данного параметра в запросе.

**“?” “=”**  **“&”**

Если посмотреть в строку ввода то при этом после End point-a в URL ставится **“?”** знак вопроса и пишется название параметра затем

**“=”** знак равно и указывается значение параметра. Всё, что идёт до знака вопроса НЕ МОЖЕТ быть query params. Если указывается несколько query parameters то они перечисляются через амперсанд **“&”**

Также существуют query parameters with multiple values.

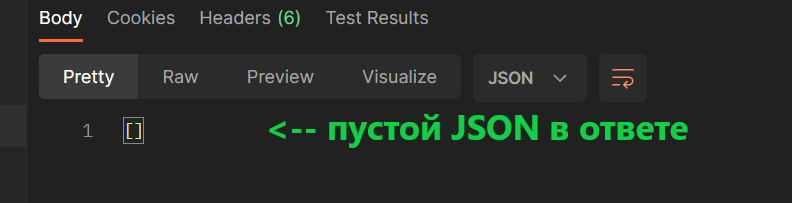
Пример: …io/postComplexJsonObject?foo\_1=**bar\_1,bar\_2,bar\_3 ←**

Пример: {{Base\_URL\_Grocery}}/products??category=fresh-produce**&**results=5

В вкладке Params это делать намного удобней, не надо набирать вручную в строке поиска и помнить про синтаксис, также активизировать/деактивизировать при необходимости проще.

К примеру у нас есть риквест с определенным набором Query Parametersи мы хотим создав новый риквест перенести все, или часть данных параметров в наш новый реквест, можно сделать это быстро, а не таскать по очереди, для этого справа в “шапке” таблицы для редактирования есть вкладка “Bulk edit” нажав на неё можно скопировать нужные пары (ключ значение), зайдя в нужный риквест обратно-же нажать на Bulk edit и вставить скопированное ранее. Затем нажать Key-value edit.

**Важно!** При написании имени параметра необходимо соблюдать табуляцию (заглавные/прописные буквы), так как данное поле является key sensitive. category и Сategory это с точки зрения Postman-a два разных параметра. Тоже самое касается и значений параметров, они должны быть написаны идентично как в документации. Most APIs are sensitive about how query parameters are written. So generally-speaking, query parameters are case-sensitive. Some APIs may handle query parameters as case-insensitive because it is more convenient and less error-prone. But this is not a rule. You should always check the API documentation to see how the query parameters are written.

Как особенность применения параметров, в респонсе можно получить пустой ответ, при этом название параметра, значение, могут быть корректными и статус код будет (200 ok), дело в том что мы получили пустой JSON, такое бывает. И это тоже важная информация говорящая, что в этой категории товаров нет, но технически всё работает исправно.

This is a representation of an empty list of data objects in JSON. This is not considered an error and the status code should be 200 OK.

“Подгружать” параметры можно только те которые указаны в документации. К примеру я получил какой то общий список продуктов и хочу отфильтровать. Я вижу, что у каждого объекта JSON-а есть поля, но это ошибка думать что по этим полям(по любому из полей) можно сделать фильтрацию и получить какую то специфическую группу продуктов (объектов), если разработчиком не предусмотрено такое поведение, то это Error.

Также если пытатся к примеру получить лист продуктов с большим/меньшим числом чем то что заявлено в документации.

Также если API не предусмотрено получение в одном запросе сразу нескольких значений одного параметра (пытаемся получить один список с элементами у которых значение поля “category” равно dairy или eggs: products?category=dairy&category=eggs) то ничего не будет, поведение API нельзя запрограммировать, это делает разработчик, мы проверяем работает ли оно следуя документации. Every API is designed and built differently. This is why it is essential to study the API documentation and to understand what is the intended usage of the API. Each API works differently. Just by looking at the response body, we can't tell which parameters does the API accept.

Query parameters are present only in the request. The response typically is an object with properties. You may find query parameters that have the same name as a property from the body, but this is not mandatory.

**!!!!!!!!!** Работая с значениями параметров стоит отметить, что если в документации описан к примеру тип значения int, но при этом мы вводим символы(char) или String+int или float(2.5) и получаем от сервера какой то внятный респонс со статусом кода 200 то это является основанием чтобы задать вопрос разработчику о корректности такого поведения (ведь это баг). Так же к этому относится когда сервер возвращает неправильное количество элементов, к примеру по документации в респонсе количество продуктов может быть запрошено 1-20, а мы запрашиваем 0, и получаем 20, или запрашиваем 21, а получаем 20, это всё потенциальные баги.