

Using HTTP request headers, you can communicate with CouchDB. Through these requests we can retrieve data from the database, store data in to the database in the form of documents, and we can view as well as format the documents stored in a database.

HTTP Request Formats

While communicating with the database we will use different request formats like get, head, post, put, delete, and copy. For all operations in CouchDB, the input data and the output data structures will be in the form of JavaScript Object Notation *JSON* object.

Following are the different request formats of HTTP Protocol used to communicate with CouchDB.

- **GET** – This format is used to get a specific item. To get different items, you have to send specific url patterns. In CouchDB using this GET request, we can get static items, database documents and configuration, and statistical information in the form of JSON documents *inmostcases*.
- **HEAD** – The HEAD method is used to get the HTTP header of a GET request without the body of the response.
- **POST** – Post request is used to upload data. In CouchDB using POST request, you can set values, upload documents, set document values, and can also start certain administration commands.
- **PUT** – Using PUT request, you can create new objects, databases, documents, views and design documents.
- **DELETE** – Using DELETE request, you can delete documents, views, and design documents.
- **COPY** – Using COPY method, you can copy documents and objects.

HTTP Request Headers

HTTP headers should be supplied to get the right format and encoding. While sending the request to the CouchDB server, you can send Http request headers along with the request. Following are the different Http request headers.

- **Content-type** – This Header is used to specify the content type of the data that we supply to the server along with the request. Mostly the type of the content we send along with the request will be MIME type or JSON *application/json*. Using Content-type on a request is highly recommended.
- **Accept** – This header is used to specify the server, the list of data types that client can understand, so that the server will send its response using those data types. Generally here, you can send the list of MIME data types the client accepts, separated by colons.

Though, using Accept in queries of CouchDB is not required, it is highly recommended to ensure that the data returned can be processed by the client.

Response Headers

These are the headers of the response sent by the server. These headers give information about the content send by the server as response.

- **Content-type** – This header specifies the MIME type of the data returned by the server. For most request, the returned MIME type is text/plain.
- **Cache-control** – This header suggests the client about treating the information sent by the server. CouchDB mostly returns the must-revalidate, which indicates that the information should be revalidated if possible.

- **Content-length** – This header returns the length of the content sent by the server, in bytes.
- **Etag** – This header is used to show the revision for a document, or a view.

Status Codes

Following is the tabular form of the status code sent by the http header and the description of it.

200 – OK	This status will be issued when a request completed successfully.
201 – Created	This status will be issued when a document is created.
202 – Accepted	This status will be issued when a request is accepted.
404 – Not Found	This status will be issued when the server is unable to find the requested content.
405 – Resource Not Allowed	This status is issued when the HTTP request type used is invalid.
409 – Conflict	This status is issued whenever there is any update conflict.
415 – Bad Content Type	This status indicated that the requested content type is not supported by the server.
500 – Internal Server Error	This status is issued whenever the data sent in the request is invalid.

HTTP URL Paths

There are certain url paths using which, you can interact with the database directly. Following is the tabular format of such url paths.

URL	Operation
PUT /db	This url is used to create a new database.
GET /db	This url is used to get the information about the existing database.
PUT /db/document	This url is used to create a document/update an existing document.
GET /db/document	This url is used to get the document.
DELETE /db/document	This url is used to delete the specified document from the specified database.
GET /db/_design/design-doc	This url is used to get the definition of a design document.
GET /db/_design/designdoc/_view/view-name	This url is used to access the view, view-name from the design document from the specified database.