1. **API’s:** Everything available to us are in the form of API’s like jar files,exe,dll. API’s are some prewritten code / business logic available to us for use. But these are deskop APIs.
2. **Web API**- API’s which are accessed over network are WEB API’s.
3. Web API’s are also called – **Web Services**.
4. **Web Service :** Its is a communication between two applications over the network via some common language a XML/JSON. (The two application may be on different languages)
5. All web services are API’s. All API”S are not webservices.
6. Web service is State-less in nature. As it uses http. As HTTP is stateless there for webservice is also stateless.
7. By stateless we mean that the connection between the server and browser ends as soon as the transaction ends.
8. **Imp**: **How can we return 404 Error** – Give a wrong URL , Or the name of the route in the url.
9. **Status Code for all uncaught exceptions** : 500 Internal Server error.
10. Web API supports which protocol? HTTP protocol.ds
11. **Content-Negotiation:**

* Content-Type- This tells the server hum kis format ka data bhejenge
* Accept-This tells the client ki type ka data accept krta hai

1. **Media-Type Formatter-**

* It is an abstract class that has two derived classes-
* JSONMediaType formatter-(for json format)
* XMLMediaType Formatter-(for xml formatter)
* These classes are responsible for serializing the response data in the format the client asked for.

1. **APIController – API controller is the controller inherited**

**Importance of Restful APIS/ Benefit of WebAPI over WCF.**

1. Rest is light weight. As it does not uses SOAP. Where as WCF uses SOAP.
2. WCF Required more configurations where as REST is simple to use using HTTP verbs
3. We can make WCF service as restful but it requires lot of configurations.,

**SOAP / Rest**

1. **SOAP-** Simple Object Access Protocal.

* It is a protocol.
* It is xml passing msgs based protocol.
* It support HTTP and other transport protocols like UDP ,TCP etc
* Supports only xml format for transmission of data.
* Soap uses SSL and WS-security
* **It works by sending an ENVELOPE**
* Business logic is hidden
* **More secure**
* **Banking firms uses.**
* **It follows strict standards.**
* Heavy , Overhead is high there for **more** **bandwidth.**

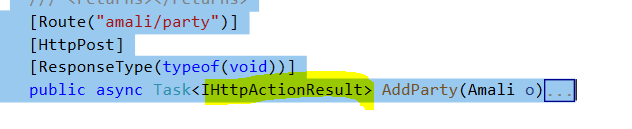
1. **Rest-** Representational State Transfer.

* It is an architecture/ design pattern.
* API which used HTTP Verbs are Restful in nature
* It uses HTTP protocoal for CRUD Opertaions.
* Create – POST
* Read -Get
* Update-PUT(updates the entire record)
* Delete- Delete
* Patch- Update on column of a record/row.
* Business logic is clear and exposed by looking at it’s URI.
* Permits both XML/ JSON for data transmission.
* Light Weight as it sends in the form of a Postcard
* **HTTP Request types**
* Create – POST
* Read -Get
* Update-PUT(updates the entire record)
* Delete- Delete
* Patch- Update on column of a record/row.
* **HTTP Response Types**

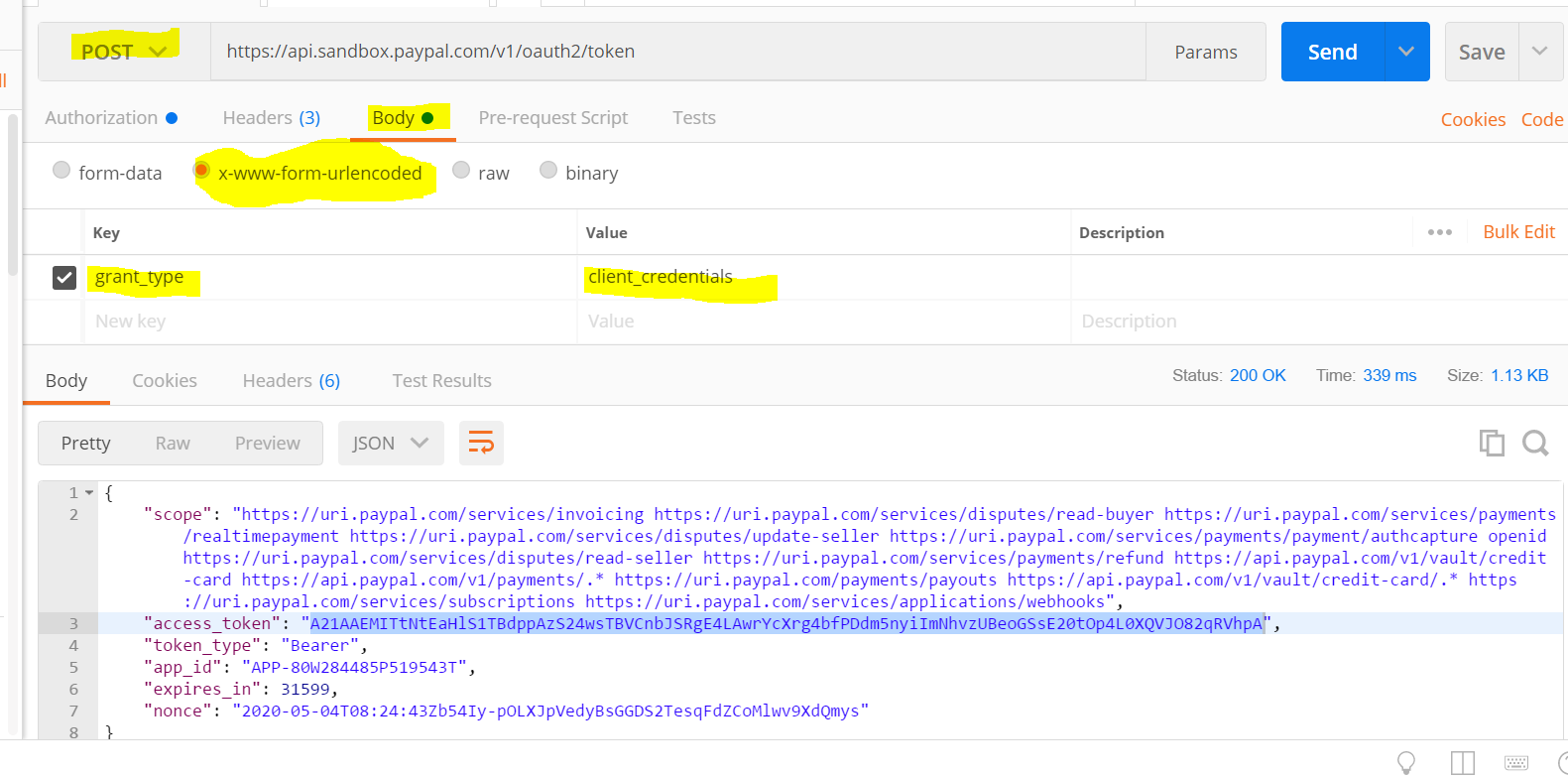
| **Status code** | **Description** |
| --- | --- |
| 200 OK | The request succeeded. |
| 201 Created | A POST method successfully created a resource. If the resource was already created by a previous execution of the same method, for example, the server returns the HTTP 200 OK status code. |
| 202 Accepted | The server accepted the request and will execute it later. |
| 204 No Content | The server successfully executed the method but returns no response body. |
| 400 Bad Request | INVALID\_REQUEST. Request is not well-formed, syntactically incorrect, or violates schema. |
| 401 Unauthorized | AUTHENTICATION\_FAILURE. Authentication failed due to invalid authentication credentials. |
| 403 Forbidden | NOT\_AUTHORIZED. Authorization failed due to insufficient permissions. |
| 404 Not Found | RESOURCE\_NOT\_FOUND. The specified resource does not exist. |
| 405 Method Not Allowed | METHOD\_NOT\_SUPPORTED. The server does not implement the requested HTTP method. |
| 406 Not Acceptable | MEDIA\_TYPE\_NOT\_ACCEPTABLE. The server does not implement the media type that would be acceptable to the client. |
| 415 Unsupported Media Type | UNSUPPORTED\_MEDIA\_TYPE. The server does not support the request payload’s media type. |
| 422 Unprocessable Entity | UNPROCESSABLE\_ENTITY. The API cannot complete the requested action, or the request action is semantically incorrect or fails business validation. |
| 429 Unprocessable Entity | RATE\_LIMIT\_REACHED. Too many requests. Blocked due to rate limiting. |
| 500 Internal Server Error | INTERNAL\_SERVER\_ERROR. An internal server error has occurred. |
| 503 Service Unavailable | SERVICE\_UNAVAILABLE. Service Unavailable. |

|  |  |
| --- | --- |
| HTTP Status Code | Description |
| 1xx | Informational |
| 2xx | Success |
| 3xx | Redrectional |
| 4xx | Client Side Error |
| 5xx | Server-side Exceptions |

* **HTTP Return types**
* Main return type supported in web api2?
* Void
* HttpResponseMessage
* IHttpActionResult

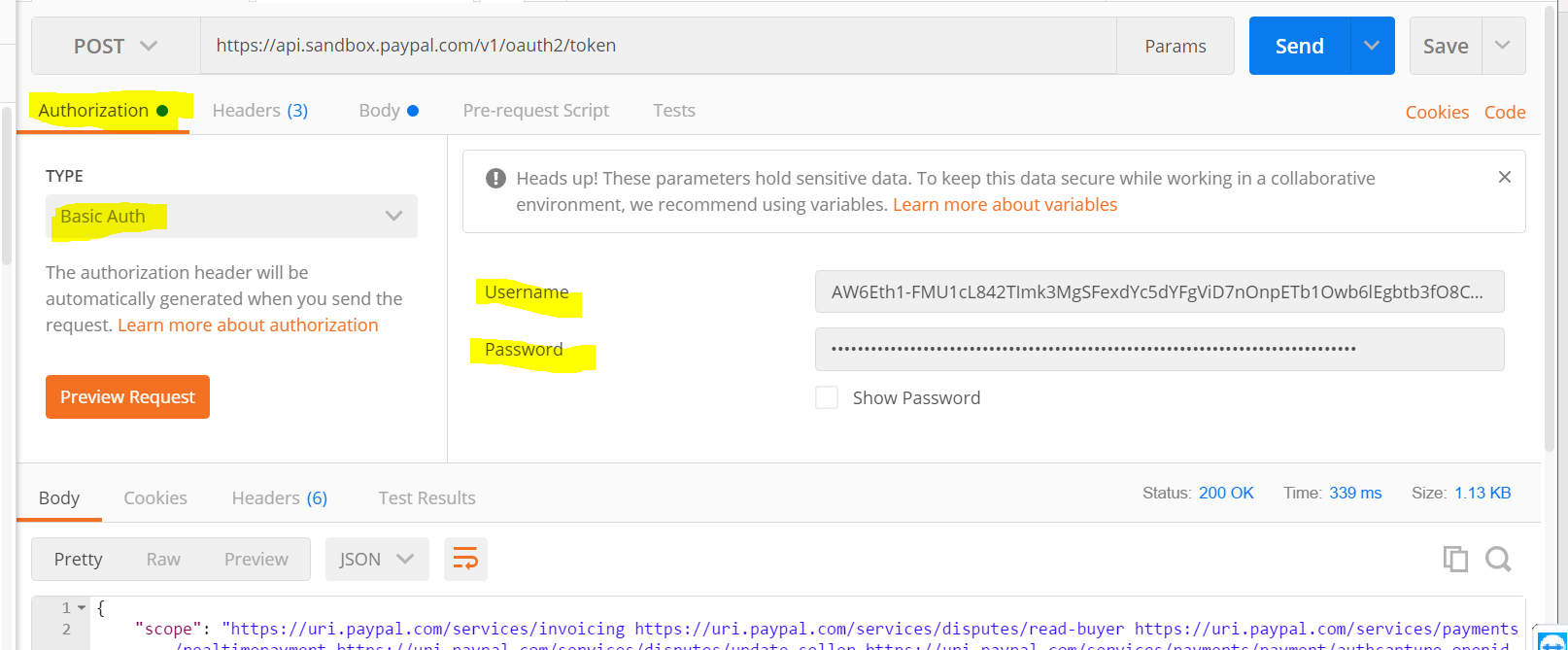


The below highlight are the content Type:

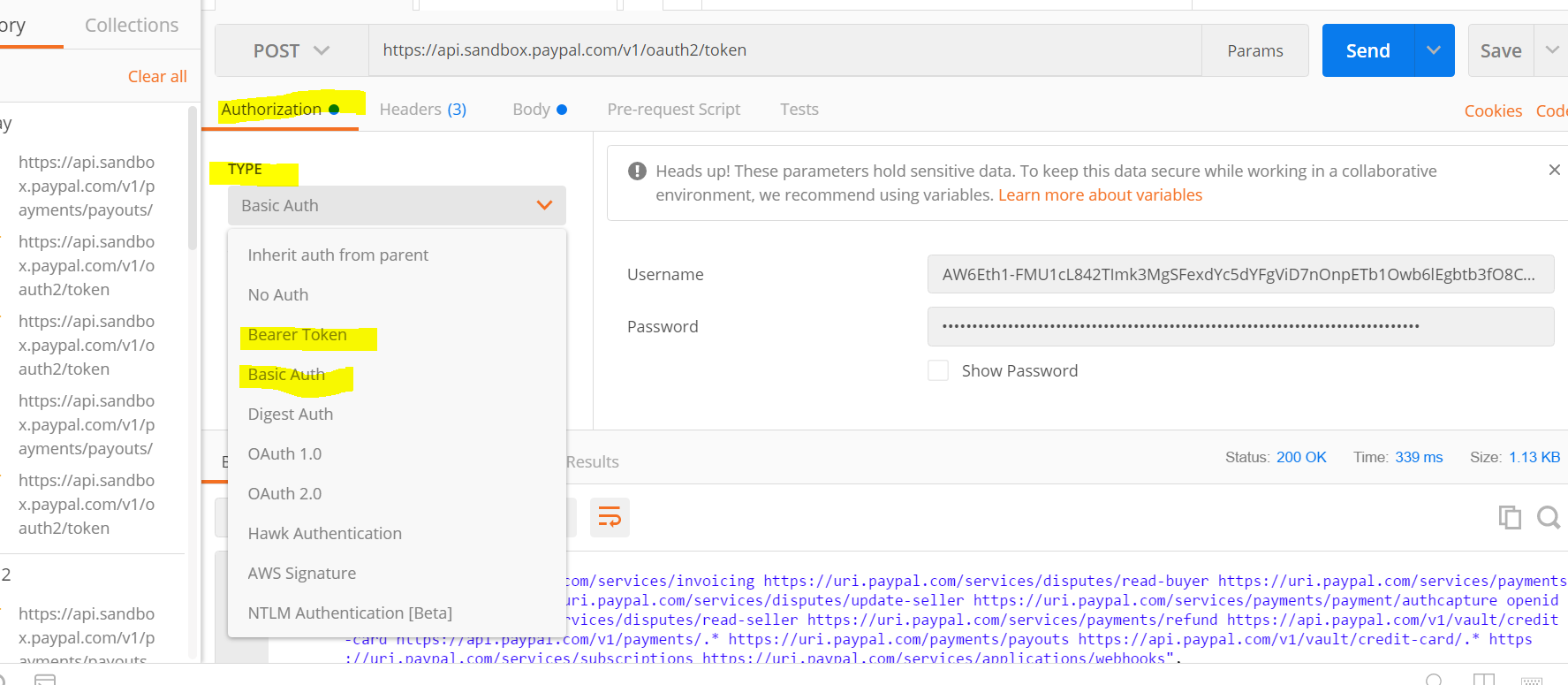


For Making a hit to Token generation API:

We provided username / password as **Basic Auth**



All the Authorization Types:



Response

