**ODS REST Service Instructions**

Examples in this documentation use the POSTMAN Google extension, a REST client which can be obtained at:

[**https://chrome.google.com/webstore/detail/postman-rest-client/fdmmgilgnpjigdojojpjoooidkmcomcm?hl=enAuthentication**](https://chrome.google.com/webstore/detail/postman-rest-client/fdmmgilgnpjigdojojpjoooidkmcomcm?hl=enAuthentication)

**Authentication**

In order to query data using the ODS REST service API you must first obtain a security token. This token will be passed via the HTTP header for all subsequent API calls. Currently, the service is configured so that the token will expire in 12 hours, after which a new security token must be generated.



Select POST passing the following data for authentication: “grant\_type=password&username=debug%40example.com&password=Debug%23123” and click Send. The service will return with the following:



Copy and paste the token into an editor, as it will be needed for subsequent calls to the service.

**Querying Data**

To get data back from the service pass the token through the http header. Click the headers button. Choose Authorization as the header type and “Bearer” plus the token obtained in the previous step. Choose GET as the HTTP option.



To obtain data it is important to structure the URL using a route that the service understands. Typically, the route will be {Server}/api/{RootRoute}/{Subroute}. In this example the server is ‘localhost’, the root route is ‘api/Invoice’, and the sub-route is the ID of the Invoice.

<http://localhost:2031/api/Invoice/D28DF62F-03A5-E111-863C-000C29BC691C>

Click ‘Send’ and the data will be returned in a JSON format, and the return status upon success will be ‘200 OK’.



**404 Not Found**

If data does not exist given the parameters sent to the service ‘404 Not Found’ will be returned.



In this case data does not exist for the sub-route ‘InvoiceLineItems/{PatientID}/{Year}, so a ‘404 Not Found’. This is to be differentiated for ‘404 Not found’ when a route is passed that the service does not recognize. For example if I use the route ‘http://localhost:2031/api/Invoice/InvoiceLineItem/0825723E-31FF-DE11-846F-001A64712B91/2014’, leaving the ‘s’ off of ‘InvoiceLineItem’ the following result will be returned:

