

Billing Report

URL:

PostURL : <http://api.audiencelogy.com/adxadminapiv4/controls/billingReport>

NOTE : TO CALL THIS API YOU NEED TO PASS THREE FIELDS IN REQUEST HEADER

1. **APIKey** : You can create this from admin interface for particular user.Key type should be **APIAccess**
2. **SessionId** : Pass 0
3. **Content-Type** : **application/json**

Request Parameter

1. **Start_Date**: specify date in format yyyy-mm-dd
2. **End_Date** : specify date in format yyyy-mm-dd
3. **Interval** : can have one of the following values {"None","Daily"}
4. **Dimension**: can have select combination of following values
{source,campcode}
5. **Filters** :
 - -1 : if you don't want to apply any filter
 - campcode : if you want to filter by campcode then pass campcode value e.g campcode = 10
6. **Imp_Start** : impression start range
7. **Imp_End** : Impression end range

Sample Request:

```
{
  "Start_Date": "2018-05-25",
  "End_Date": "2018-05-25",
  "Interval": "None",
  "Dimension": "campcode",
  "Filters": "-1",
  "Imp_Start": "1",
  "Imp_End": "1000000000"
}
```

Sample Response

```
{
  "ReturnCode": "0",
  "Message": "Success",
  "Data": [{
    "impressions": 0,
    "clicks": 0,
    "actions": 0,
    "ctr": 0,
    "rpm": 0,
    "rpc": 0,
    "rpa": 0,
    "campaign_name": "Test",
    "buyer_spend": 0.00000,
    "campcode": 818
  }]
}
```

Transaction Research Report

URL:

PostURL: <http://api.audienceology.com/adxadminapiv4/controls/getControlInformation>

NOTE : TO CALL THIS API YOU NEED TO PASS THREE FIELDS IN REQUEST HEADER

1. **APIKey** : You can create this from admin interface for particular user.
Key type should be **APIAccess**.
2. **SessionId** : Pass 0
3. **Content-Type** : **application/json**

Request Params:

1. **Start_Date** : specify date in format yyyy-mm-dd
2. **End_Date** : specify end date in format yyyy-mm-dd
3. **Interval**: can have one of following values {"None","Daily","Hourly"}
If "Interval"="Hourly" then in request you have to pass 2 additional parameter "Start_Hour" , "End_Hour" with specific hour interval
4. **Dimension** : You can select combination of camp_code,banner_code with any one from
("referrer","audience","country","external_site_id","browser","mcc","OS")
e.g. 1. camp_code,country
e.g. 2. camp_code,audience
5. **Filters** : can have following values
 - -1 : if you don't want to apply any filter
 - Source=10 : if you want to apply filter by source
 - camp_code =101 :if you want to apply filter by campcode
6. **Imp_Start**: impression start range
7. **Imp_End**: impression end range

Sample request:

```
{
  "Start_Date": "2018-05-17",
  "End_Date": "2018-05-24",
  "Interval": "Hourly",
  "Dimension": "camp_code,referrer",
  "Filters": "source in (3)",
  "Imp_Start": "1",
  "Imp_End": "1000000001",
  "Start_Hour": "1",
  "End_Hour": "8"
}
```

Sample response

```
{
  "ReturnCode": "0",
  "Message": "Success",
  "Data": [{
    "referrer": "droom.sleepIfUCan",
    "campaign_name": "Test",
    "camp_code": 7605,
    "impressions": 3279,
    "clicks": 0,
    "actions": 0,
    "ctr": 0,
    "rpm": 0,
    "rpc": 0,
    "rpa": 0,
    "spending": 0.84950
  }]
}
```

SAMPLE CODE TO CALL API (.net)

```
XmlDocument doc = new XmlDocument();
HttpWebRequest req = (HttpWebRequest)WebRequest.Create(<API link>);
req.ProtocolVersion = HttpVersion.Version10;
req.Proxy = WebRequest.DefaultWebProxy;
req.Accept = "application/json";
req.Method = "POST";
req.Headers.Add("APIKEY", <API Key>);
req.Headers.Add("SessionId", "0");

// Create POST data and convert it to a byte array.
byte[] byteArray = Encoding.UTF8.GetBytes(<Json String>);

// Set the ContentType property of the WebRequest.
req.ContentType = "application/x-www-form-urlencoded";

// Set the ContentLength property of the WebRequest.
req.ContentLength = byteArray.Length;

// Get the request stream.
Stream stm = req.GetRequestStream();

// Write the data to the request stream.
stm.Write(byteArray, 0, byteArray.Length);

// Close the Stream object.
stm.Close();

WebResponse resp = req.GetResponse();
stm = resp.GetResponseStream();
StreamReader r = new StreamReader(stm);
string ReturnXML = r.ReadToEnd();
req.KeepAlive = false;
```