

API Inquiry: `get_eventgroup2a`

| | | | |
|---------------------------|---|-----------------|---|
| Description | Given an event code and group transnum parameter, returns a single event group (team or table) record. | | |
| Parameters | scheme | GET | <i>Required.</i> All connection types. Inquiry type indicator. |
| | action | get_eventgroup2 | <i>Required.</i> All connection types. Variable representing the specific inquiry. |
| | URL | nnn.nnn.n.n | <i>Optional.</i> Direct and DAD only. The URL of the requester, used for reporting. Results will display only if a URL is included. |
| | dsource | xxxxxxx | <i>Required.</i> Connection pooling servlet only. The key to the client door to the database. |
| | datafile | event | <i>Required.</i> Restricts the event groups (teams or tables) selected to those belonging the event. |
| | | transnum | <i>Required.</i> Restricts to a single event group (team or table). |
| Sample Data | <pre><?xml version="1.0" encoding="UTF-8"?> <XMLREQUEST> <RECORD> <EVENT>TESTEVENT</EVENT> <TRANSNUM>1234</TRANSNUM> </RECORD> </XMLREQUEST></pre> | | |
| Sample Result | <pre><?xml version="1.0" ?> <GET_EVENTGROUP2A TRANS_TIMESTAMP="09/06/2006 01:06:35 pm" EVENT="TESTEVENT" TRANSNUM="2137"> <EVENT_GROUPS> <EVENT_GROUPS.GROUPTYPE>GRPT</EVENT_GROUPS.GROUPTYPE> <EVENT_GROUPS.NUMPOSITIONS>10</EVENT_GROUPS.NUMPOSITIONS> <EVENT_GROUPS.GROUPNAME>Team B</EVENT_GROUPS.GROUPNAME> <EVENT_GROUPS.EVENT>TESTEVENT</EVENT_GROUPS.EVENT> <EVENT_GROUPS.TIMESTAMP>2006-09-05</EVENT_GROUPS.TIMES TAMP><EVENT_GROUPS.TRANSNUM>2137</EVENT_GROUPS.TRANSNUM> <EVENT_GROUPS.USERID_ADD>TESTV440</EVENT_GROUPS.USERID_ADD> </EVENT_GROUPS> </GET_EVENTGROUP2A></pre> | | |
| Additional Reading | Events, relevant API connection documents, <code>get_eventgroup1</code> , <code>get_eventgroup1a</code> , <code>get_eventgroup2</code> | | |
| Reference | E430092 | | |

API Inquiry: *get_eventgroup2a*

| | |
|------------------------------|--|
| Connection Pooling | http://webapps.pledgemaker.com/test/proc?d_source=12345678&scheme=GET&action=get_evengroup2a&datafile= |
| Data Access Descriptor (DAD) | <a href="http://pm-devel/pls/testv440/apiwrap.h_traffic?scheme=GET&url=128.168.0.1&action=xml_pm.get_eventgroup2a&data=<?xml%version='1.0'%encoding='UTF-8'?><XMLREQUEST><RECORD><EVENT>TESTEVENT</EVENT><TRANSNUM>2137</TRANSNUM></RECORD></XMLREQUEST>">http://pm-devel/pls/testv440/apiwrap.h_traffic?scheme=GET&url=128.168.0.1&action=xml_pm.get_eventgroup2a&data=<?xml%version='1.0'%encoding='UTF-8'?><XMLREQUEST><RECORD><EVENT>TESTEVENT</EVENT><TRANSNUM>2137</TRANSNUM></RECORD></XMLREQUEST> |
| Direct Connection | <pre> set serveroutput on; declare vScheme varchar2(100) := 'GET'; vAction varchar2(200) := 'xml_pm.get_eventgroup2a'; vURL varchar2(100) := '128.168.0.1'; vXML varchar2(10000); rval_clob CLOB; begin owa.num_cgi_vars := 0; vXML := '<?xml version="1.0" encoding="UTF-8"?>' CHR(10) ' <XMLREQUEST>' CHR(10) ' <RECORD>' CHR(10) ' <EVENT>TESTEVENT</EVENT>' CHR(10) ' <TRANSNUM>2137</TRANSNUM>' CHR(10) ' </RECORD>' CHR(10) ' </XMLREQUEST>'; rval_clob := NULL; apiwrap.all_traffic(vScheme, vAction, vURL, vXML, rval_clob); IF (rval_clob IS NOT NULL) THEN dbms_output.put_line(substr(rval_clob,1,200)); dbms_output.put_line(substr(rval_clob,201,200)); dbms_output.put_line(substr(rval_clob,401,200)); dbms_output.put_line(substr(rval_clob,601,200)); dbms_output.put_line(substr(rval_clob,801,200)); END IF; end; </pre> |