

Advanced Query to get Top Event

01/10/2024 1:46 pm EST

This script uses several techniques that are useful to know when doing these types of queries. Please see the Advanced Query to get top event attachment for the SQL.

Premise:

Manitou needs a single result returned from the SQL query. Additionally, we need the second result returned because we need to know what happened in the past. We are looking to see if a *EM or a *LT happened in the last 96 outside of the event that just came in. We also have to check this month and last month because the alarm could have come in between last month's activity log and this month's activity log when crossing from the last day of the month to the first day of the month.

Formatted for Manitou:

SET: @TEST =

```
BEGIN;DECLARE @THIS_MONTH NVARCHAR(8); DECLARE @LAST_MONTH NVARCHAR(8); SET @THIS_MONTH=(SELECT STCODE FROM CLOG{0} WHERE STCODE IN ("*LT", "*EM") AND EVTYPE IN (1,0) AND SERIALNO={ME} AND LOGDATE>=DATEADD(HH,-96,"{DU}") ORDER BY CREDATE DESC OFFSET 1 ROWS FETCH NEXT 1 ROW ONLY); SET @LAST_MONTH=(SELECT STCODE FROM CLOG{1} WHERE STCODE IN ("*LT", "*EM") AND EVTYPE IN (1,0) AND SERIALNO=24 AND LOGDATE>=DATEADD(HH,-96,"{DU}") ORDER BY CREDATE DESC OFFSET 1 ROWS FETCH NEXT 1 ROW ONLY); IF @THIS_MONTH IS NOT NULL SELECT @THIS_MONTH; ELSE BEGIN IF @LAST_MONTH IS NOT NULL SELECT @LAST_MONTH; ELSE SELECT "NONE"; END; END;
```

Please note this is set to offset by one so it will get the second to top item. The query in the comments has the offset set to 0 and gets the top row.



```
SET: @CHKUSRNO = BEGIN;DECLARE @THIS_MONTH NVARCHAR(8); DECLARE @LAST_MONTH NVARCHAR(8); SET @THIS_MONTH =(SELECT USRNO FROM CLOG{0} WHERE STCODE = "OP" AND EVTYPE IN (1,0) AND SERIALNO={ME} AND LOGDATE>=DATEADD(MINUTE,-30, GETUTCDATE()) ORDER BY CREDATE DESC OFFSET 0 ROWS FETCH NEXT 1 ROW ONLY); SET @LAST_MONTH =(SELECT USRNO FROM CLOG{1} WHERE STCODE = "OP" AND EVTYPE IN (1,0) AND SERIALNO={ME} AND LOGDATE>=DATEADD(MINUTE,-30, GETUTCDATE()) ORDER BY CREDATE DESC OFFSET 0 ROWS FETCH NEXT 1 ROW ONLY); IF @THIS_MONTH IS NOT NULL SELECT @THIS_MONTH ; ELSE BEGIN IF @LAST_MONTH IS NOT NULL SELECT @LAST_MONTH ; ELSE SELECT "NONE"; END; END; LOG: USER @CHKUSRNO OPENED LAST. - SELECT @CHKUSRNO - CASE = 1 CONTACT CUSTOMER USING CALLLIST OC1 - CASE = 2 CONTACT CUSTOMER USING CALLLIST OC2 - CASE = 3 CONTACT CUSTOMER USING CALLLIST OC3 - OTHERWISE CONTACT CUSTOMER USING CALLLIST RESP END SELECT
```

Comments

Deleting comment...

What I did for JD security: BEGIN; DECLARE @THIS_MONTH NVARCHAR(8); DECLARE @LAST_MONTH NVARCHAR(8); SET @THIS_MONTH=(SELECT USRNO FROM CLOG{0} WHERE STCODE = "OP" AND EVTYPE IN (1,0) AND SERIALNO={ME} AND LOGDATE>=DATEADD(MINUTE,-30,GETUTCDATE())) ORDER BY CREDATE DESC OFFSET 0 ROWS FETCH NEXT 1 ROW ONLY); SET @LAST_MONTH=(SELECT USRNO FROM CLOG{1} WHERE STCODE = "OP" AND EVTYPE IN (1,0) AND SERIALNO={ME} AND LOGDATE>=DATEADD(MINUTE,-30,GETUTCDATE())) ORDER BY CREDATE DESC OFFSET 0 ROWS FETCH NEXT 1 ROW ONLY); IF @THIS_MONTH IS NOT NULL SELECT @THIS_MONTH; ELSE BEGIN IF @LAST_MONTH IS NOT NULL SELECT @LAST_MONTH; ELSE SELECT "NONE"; END; END;

What I did for JD security: BEGIN; DECLARE @THIS_MONTH NVARCHAR(8); DECLARE @LAST_MONTH NVARCHAR(8); SET @THIS_MONTH=(SELECT USRNO FROM CLOG{0} WHERE STCODE = "OP" AND EVTYPE IN (1,0) AND SERIALNO={ME} AND LOGDATE>=DATEADD(MINUTE,-30,GETUTCDATE())) ORDER BY CREDATE DESC OFFSET 0 ROWS FETCH NEXT 1 ROW ONLY); SET @LAST_MONTH=(SELECT USRNO FROM CLOG{1} WHERE STCODE = "OP" AND EVTYPE IN (1,0) AND SERIALNO={ME} AND LOGDATE>=DATEADD(MINUTE,-30,GETUTCDATE())) ORDER BY CREDATE DESC OFFSET 0 ROWS FETCH NEXT 1 ROW ONLY); IF @THIS_MONTH IS NOT NULL SELECT @THIS_MONTH; ELSE BEGIN IF @LAST_MONTH IS NOT NULL SELECT @LAST_MONTH; ELSE SELECT "NONE"; END; END;

CancelSave Comment

Saving comment...

Deleting comment...

As of 2.1.9, this can only be built in the legacy client. It errors in the syntax checker.

As of 2.1.9, this can only be built in the legacy client. It errors in the syntax checker.

CancelSave Comment

Saving comment...