

Ed-Fi 2.x Grading Period Entity Mapping

Overview

Use this page to help understand and define the mapping of Aeries data to Ed-Fi data



Status

Ready for certification

Document Version 1.5

Last Updated 25 Feb 2020

Note

If the school is an elementary school then get the data from the TRM table, else get the data from the GRP table.

Links

Entity

<https://mappingedu.ed-fi.org/#!/entity/2e87fca4-4ed2-e811-80d9-005056b36109/info?dataStandardId=db3697f4-4dd2-e811-80d9-005056b36109>

Scenarios

<https://techdocs.ed-fi.org/display/EDFICERT/Education+Organization+Calendar+%3E+Grading+Periods+Scenarios>

Schema

<https://schema.ed-fi.org/datahandbook-v22/Ed-Fi-UDM-Handbook-Index.html#/GradingPeriod561>

Elementary Schools

Root Table

TRM

SQL

Scheduled

TRM_Functions.GetEdFiDataTable_GradingPeriods

```
SELECT [SC], [D1] AS BeginDate, [D2] AS EndDate, [TM] AS StateCode, [TM] AS SEQ, 'Elementary' AS SchoolType
FROM [TRM]
WHERE [DEL] = 0 AND [SC] = @SC
```

Real-Time

TRM_Functions.GetEdFiDataTable_GradingPeriods

```
SELECT [SC], [D1] AS BeginDate, [D2] AS EndDate, [TM] AS StateCode, [TM] AS SEQ, 'Elementary' AS SchoolType
FROM [TRM]
WHERE [DEL] = 0 AND [SC] = @SC AND [DTS] >= @TimeSince
```

Secondary Schools

Root Table

GRP

SQL

Scheduled

GRP_Functions.GetEdFiDataTable_GradingPeriods

```
WITH cteGRP ([SC], [KY], [DTS]
, [B1], [B2], [B3], [B4], [B5], [B6], [B7], [B8], [B9], [B10], [B11], [B12]
, [E1], [E2], [E3], [E4], [E5], [E6], [E7], [E8], [E9], [E10], [E11], [E12]
, [C1], [C2], [C3], [C4], [C5], [C6], [C7], [C8], [C9], [C10], [C11], [C12], RowNumber) AS
(
    SELECT [SC], [KY], [DTS]
    , [B1], [B2], [B3], [B4], [B5], [B6], [B7], [B8], [B9], [B10], [B11], [B12]
    , [E1], [E2], [E3], [E4], [E5], [E6], [E7], [E8], [E9], [E10], [E11], [E12]
    , [C1], [C2], [C3], [C4], [C5], [C6], [C7], [C8], [C9], [C10], [C11], [C12]
    , ROW_NUMBER() OVER(PARTITION BY [SC] ORDER BY [KY]) [RowNumber]
    FROM [GRP]
    WHERE [DEL] = 0 AND [SC] = @SC
)
SELECT [SC], [KY], [B1] 'BeginDate', [E1] 'EndDate', 1 [SEQ], [C1] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C1] > ' '
UNION
SELECT [SC], [KY], [B2] 'BeginDate', [E2] 'EndDate', 2 [SEQ], [C2] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C2] > ' '
UNION
SELECT [SC], [KY], [B3] 'BeginDate', [E3] 'EndDate', 3 [SEQ], [C3] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C3] > ' '
UNION
SELECT [SC], [KY], [B4] 'BeginDate', [E4] 'EndDate', 4 [SEQ], [C4] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C4] > ' '
UNION
SELECT [SC], [KY], [B5] 'BeginDate', [E5] 'EndDate', 5 [SEQ], [C5] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C5] > ' '
UNION
SELECT [SC], [KY], [B6] 'BeginDate', [E6] 'EndDate', 6 [SEQ], [C6] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C6] > ' '
UNION
SELECT [SC], [KY], [B7] 'BeginDate', [E7] 'EndDate', 7 [SEQ], [C7] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C7] > ' '
UNION
SELECT [SC], [KY], [B8] 'BeginDate', [E8] 'EndDate', 8 [SEQ], [C8] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C8] > ' '
UNION
SELECT [SC], [KY], [B9] 'BeginDate', [E9] 'EndDate', 9 [SEQ], [C9] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C9] > ' '
UNION
SELECT [SC], [KY], [B10] 'BeginDate', [E10] 'EndDate', 10 [SEQ], [C10] 'StateCode', 'Secondary' AS
SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C10] > ' '
UNION
SELECT [SC], [KY], [B11] 'BeginDate', [E11] 'EndDate', 11 [SEQ], [C11] 'StateCode', 'Secondary' AS
SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C11] > ' '
UNION
SELECT [SC], [KY], [B12] 'BeginDate', [E12] 'EndDate', 12 [SEQ], [C12] 'StateCode', 'Secondary' AS
SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C12] > ' '

```

Real-Time

GRP_Functions.GetEdFiDataTable_GradingPeriods

```
WITH cteGRP ([SC], [KY], [DTS]
, [B1], [B2], [B3], [B4], [B5], [B6], [B7], [B8], [B9], [B10], [B11], [B12]
, [E1], [E2], [E3], [E4], [E5], [E6], [E7], [E8], [E9], [E10], [E11], [E12]
, [C1], [C2], [C3], [C4], [C5], [C6], [C7], [C8], [C9], [C10], [C11], [C12], RowNumber) AS
(
    SELECT [SC], [KY], [DTS]
    , [B1], [B2], [B3], [B4], [B5], [B6], [B7], [B8], [B9], [B10], [B11], [B12]
    , [E1], [E2], [E3], [E4], [E5], [E6], [E7], [E8], [E9], [E10], [E11], [E12]
    , [C1], [C2], [C3], [C4], [C5], [C6], [C7], [C8], [C9], [C10], [C11], [C12]
    , ROW_NUMBER() OVER(PARTITION BY [SC] ORDER BY [KY]) [RowNumber]
    FROM [GRP]
    WHERE [DEL] = 0 AND [SC] = @SC
)
SELECT [SC], [KY], [B1] 'BeginDate', [E1] 'EndDate', 1 [SEQ], [C1] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C1] > ' ' AND [DTS] >= @TimeSince
UNION
SELECT [SC], [KY], [B2] 'BeginDate', [E2] 'EndDate', 2 [SEQ], [C2] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C2] > ' ' AND [DTS] >= @TimeSince
UNION
SELECT [SC], [KY], [B3] 'BeginDate', [E3] 'EndDate', 3 [SEQ], [C3] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C3] > ' ' AND [DTS] >= @TimeSince
UNION
SELECT [SC], [KY], [B4] 'BeginDate', [E4] 'EndDate', 4 [SEQ], [C4] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C4] > ' ' AND [DTS] >= @TimeSince
UNION
SELECT [SC], [KY], [B5] 'BeginDate', [E5] 'EndDate', 5 [SEQ], [C5] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C5] > ' ' AND [DTS] >= @TimeSince
UNION
SELECT [SC], [KY], [B6] 'BeginDate', [E6] 'EndDate', 6 [SEQ], [C6] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C6] > ' ' AND [DTS] >= @TimeSince
UNION
SELECT [SC], [KY], [B7] 'BeginDate', [E7] 'EndDate', 7 [SEQ], [C7] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C7] > ' ' AND [DTS] >= @TimeSince
UNION
SELECT [SC], [KY], [B8] 'BeginDate', [E8] 'EndDate', 8 [SEQ], [C8] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C8] > ' ' AND [DTS] >= @TimeSince
UNION
SELECT [SC], [KY], [B9] 'BeginDate', [E9] 'EndDate', 9 [SEQ], [C9] 'StateCode', 'Secondary' AS SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C9] > ' ' AND [DTS] >= @TimeSince
UNION
SELECT [SC], [KY], [B10] 'BeginDate', [E10] 'EndDate', 10 [SEQ], [C10] 'StateCode', 'Secondary' AS
SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C10] > ' ' AND [DTS] >= @TimeSince
UNION
SELECT [SC], [KY], [B11] 'BeginDate', [E11] 'EndDate', 11 [SEQ], [C11] 'StateCode', 'Secondary' AS
SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C11] > ' ' AND [DTS] >= @TimeSince
UNION
SELECT [SC], [KY], [B12] 'BeginDate', [E12] 'EndDate', 12 [SEQ], [C12] 'StateCode', 'Secondary' AS
SchoolType
FROM cteGRP
WHERE RowNumber = 1 AND [C12] > ' ' AND [DTS] >= @TimeSince
```

Entity Mapping

Parent	Properties	Type	Required	Table	Field	Other	Code / Notes
\	beginDate	Date	Identity	GRP	BeginDate		GetNullableDate(s("BeginDate"))
\	endDate	Date	Required	GRP	EndDate		GetNullableDate(s("EndDate"))
\	periodSequence	Integer	Required		SEQ 1-12		GetNullableInteger(s("SEQ"))
\	descriptor	GradingPeriod MapType	Identity	GRP	StateCode		GetEdFiCodeValue(GradingPeriodDescriptor, "GRP", "C1", s("StateCode"))
\	schoolReference	SchoolReference	Identity				
schoolReference	schoolReference. schoolId	Integer	Identity	GRP	SC		GetNullableInteger(s("SC"))
\	totalInstructional Days	Integer	Required	GRP	SC, BeginDate, EndDate		GetTotalInstructionalDays(drGRP!SC, drGRP!BeginDate, drGRP!EndDate) see function below

Code

```
Public Shared Function GetTotalInstructionalDays(schoolCode As Integer, startDate As Date, endDate As Date,
ByVal connectionString As String) As Integer?
```

```
Return DAY_Functions.GetTeachingDays(schoolCode, "", startDate, endDate, connectionString)
```

```
End Function
```

```
Public Shared Function GetTeachingDays(ByVal SC As Integer, ByVal TR As String, ByVal StartDate As Date,
ByVal EndDate As Date, ByVal ConnectionString As String) As Integer
```

```
Dim intDays As Integer = 0
```

```
' Ensure Date-Range is defined
```

```
If Not StartDate = Nothing AndAlso IsDate(StartDate) AndAlso Not EndDate = Nothing AndAlso IsDate
(EndDate) Then
```

```
Dim strSQL As String = "SELECT DY, DT, HO, T1, T2, T3, T4, T5, T6, T7, T8, T9, T10, T11,
T12, T13,"
```

```
strSQL &= " T14, T15, T16, T17, T18, T19, T20, T21, T22, T23, T24, T25, T26"
```

```
strSQL &= " FROM DAY WHERE DEL = 0 AND SC = @SC AND DT >= @SD AND DT <= @ED ORDER BY DY"
```

```
Dim intTR As Integer = 0
```

```
If TR > " " Then
```

```
intTR = Asc(UCase(TR)) - 64
```

```
If intTR > 26 OrElse intTR < 1 Then intTR = 0
```

```
End If
```

```
Dim drDAY As SqlDataReader = ConnectDB.GetDataReader(strSQL, ConnectionString, New
SqlParameter() {
```

```
MakeParam("@SC", SqlDbType.SmallInt, 0, SC), MakeParam("@SD", SqlDbType.Date, 0,
StartDate), MakeParam("@ED", SqlDbType.Date, 0, EndDate)})
```

```
While drDAY.Read
```

```
If intTR > 0 AndAlso intTR <= 26 Then
```

```
If isSchoolDay(drDAY("T" & intTR)) Then intDays += 1
```

```
Else
```

```
If isSchoolDay(drDAY!HO) Then intDays += 1
```

```
End If
```

```
End While
```

```
drDAY.Close()
```

```
End If
```

```
Return intDays
```

```
End Function
```