Student Pass-Through Queries

Last Modified on 08/05/2026 3:20 pm CDT

Tool Search: Filter Designer

The following queries pertain to student information and require that the Student radio button be selected on the main Filter Designer page. Student pass-through queries inherit the year, school, calendar, and schedule selected in the Campus toolbar.

These sample queries are for example purposes only. They serve as a starting point to illustrate how this tool can be used. A request for Campus Support to design, modify, or troubleshoot a Pass-Through Query is a request for Technical Services, which is subject to a fee.

Student Queries for Census and Demographic Data

Student Birth Date from Range

This query returns students whose birth dates fall within a specified time range. It includes students with birth dates that occur on the specified start or end date.

Box A

```
No text necessary
```

Box B

```
AND student.birthdate BETWEEN 'MM/DD/YYYY' AND 'MM/DD/YYYY'
```

Student Birthdays by Week

This query lists students with a birthday this week (Monday through Sunday).

Box A

```
No text necessary
```

Box B

```
AND student.activeyear=1
AND student.startdate<=getdate()
AND (student.enddate>=getdate() OR student.enddate is NULL)
AND DATEADD( Year,DATEPART( Year, GETDATE()) - DATEPART( Year, student.birthdate), student.birthdate)
BETWEEN dateadd(wk,datediff(wk,0,getdate()),0)
AND dateadd(wk,datediff(wk,0,getdate()),6)
```

Students without a Current Household

This query returns students who do not have a current household indicated.

Box A

```
LEFT OUTER JOIN HouseholdMember hm ON hm.personID = student.personID
AND (hm.endDate IS NULL OR hm.endDate > getDate())
```

Box B

```
AND hm.personID IS NULL
```

Students in a Household by Themselves

This query returns students in households with no other members.

Box A

```
INNER JOIN HouseholdMember hm1 ON hm1.personID = student.personID
AND (hm1.startDate IS NULL OR hm1.startDate <= getDate()) AND (hm1.endDate IS NULL OR hm1.endDate > getDat
e())
INNER JOIN HouseholdMember hm2 ON hm2.householdID = hm1.householdID
AND (hm2.startDate IS NULL OR hm2.startDate <= getDate())AND (hm2.endDate IS NULL OR hm2.endDate > getDate
())
```

Box B

```
GROUP BY student.personID, student.lastname, student.firstname, student.grade, student.studentnumber
HAVING COUNT(hm2.memberID) = 1
```

Students in Households with a Combined Total of More than Five

This query returns students from households with more than 5 members.

Box A

```
INNER JOIN HouseholdMember hm1 ON hm1.personID = student.personID
AND (hm1.startDate IS NULL OR hm1.startDate <=getDate()) AND (hm1.endDate IS NULL OR hm1.endDate >getDate
())
INNER JOIN HouseholdMember hm2 ON hm2.householdID = hm1.householdID
AND (hm2.startDate IS NULL OR hm2.startDate <= getDate ()) AND (hm2.endDate IS NULL OR hm2.endDate >
getDate())
```

Box B

```
GROUP BY student.personID, student.lastname, student.firstname, student.grade, student.studentnumber, hm1.house
holdID
HAVING COUNT(hm2.memberID) > 5
```

Students in a Household with no Address

This query returns students whose household(s) have no address assigned.

Box A

```
INNER JOIN HouseholdMember HM ON HM.personID=student.personID
INNER JOIN household H on h.householdID=HM.householdID
LEFT OUTER JOIN HouseholdLocation L on L.householdID=H.householdID
```

Box B

```
AND L.addressID IS NULL
```

Student not in any Household with a Mailing Address - Includes Secondary Addresses

*This query returns students who are not in any household with a mailing address. If a student has three addresses, where one is marked as a mailing address, but the other two are not, the student is **NOT** returned by the query.*

Box A

```
LEFT OUTER JOIN HouseholdMember hm ON hm.personID = student.personID AND (hm.endDate IS NULL OR hm.endD
ate > getDate())
LEFT OUTER JOIN HouseholdLocation hl ON hl.householdID = hm.householdID
AND (hl.endDate IS NULL OR hl.endDate > getDate())
AND hl.mailing = 1
```

Box B

```
AND hl.locationID IS NULL
```

Students not in any Household with a Mailing Address - Excludes Secondary Household and Secondary Addresses

This query returns students who are not in any household with a mailing address. If a student has

*three addresses, where one of them is marked as a mailing address, but the other two are not, the student is **NOT** returned by the query.*

Box A

```
LEFT OUTER JOIN HouseholdMember hm ON hm.personID = student.personID AND (hm.endDate IS NULL OR hm.endD
ate > getDate())
 AND (hm.secondary IS NULL OR hm.secondary = 0)
 LEFT OUTER JOIN HouseholdLocation hl ON hl.householdID = hm.householdID
 AND (hl.endDate IS NULL OR hl.endDate > getDate())
 AND (hl.secondary IS NULL OR hl.secondary = 0) AND hl.mailing = 1
```

Box B

```
AND hl.locationID IS NULL
```

Students with More Than One Primary Mailing Address

This query returns students with more than one primary mailing address. This query pulls from the Student View and considers the endYear. Students who have more than one enrollment in multiple schools in the school year report twice.

Box A

```
INNER JOIN HouseholdMember hm ON hm.personID = student.personID
AND (hm.endDate IS NULL OR hm.endDate > getDate()) AND (hm.secondary IS NULL OR hm.secondary = 0)
INNER JOIN HouseholdLocation hl ON hl.householdID = hm.householdID
AND (hl.endDate IS NULL OR hl.endDate > getDate())AND (hl.secondary IS NULL OR hl.secondary = 0) AND hl.mailing
= 1
```

Box B

```
GROUP BY student.personID, student.lastName, student.firstName, student.grade, student.studentNumber HAVING C
OUNT(hl.locationID) > 1
```

Student Queries for Course Information

Students who did not make a course request for a certain type of course by Course Group

This query returns a list of students who did not make a course request for a specific course type, such as Language Arts or Math. Modify the course group name (in the query) from 'Language Arts' to meet your district's needs. Ensure the query is returning the credit name, not the department name.

Box A

```
LEFT JOIN (Request r
JOIN Course crs ON crs.courseID = r.courseID
JOIN GradingTaskCredit gtc ON gtc.courseID = crs.courseID
JOIN CurriculumStandard cs ON cs.standardID = gtc.standardID And cs.name = 'Language Arts')
ON r.personID = student.personID And student.calendarID = crs.calendarID
```

Box B

```
AND crs.number IS NULL
```

Students who did not make a course request for a certain type of course by Department

This query returns a list of students who did not make a course request for a type of course, like Language Arts or Math. Modify the name of the Department (in the query as Language Arts) to meet the needs of your district.

Box A

```
LEFT JOIN (Request r join Course crs ON crs.courseID = r.courseID
JOIN Department d ON d.departmentID = crs.departmentID And d.name = 'Language Arts') ON r.personID = student.p
ersonID AND student.calendarID  = crs.calendarID
```

Box B

```
AND crs.number IS NULL
```

Students Taking a Course

This query returns students who are enrolled in a specified course. Use Box B to indicate a specific course or a general type of course (e.g., *specifying "%Literature%" pulls all course names containing the word "literature").*

Box A

```
INNER JOIN Roster r ON r.personID = student.personID
INNER JOIN Section s ON s.sectionID = r.sectionID
INNER JOIN Course c ON c.courseID = s.courseID
AND c.calendarID = student.calendarID
INNER JOIN Trial t ON t.trialID = s.trialID AND t.active = 1
```

Box B

```
AND c.name LIKE '%Literature%'
```

Students Registered for Part of a Course

This query returns students who have requested only one semester of a year-long course, rather than both semesters (e.g., if Algebra is a year-long course, students registered for Semester 1 but

not Semester 2, or vice versa, are included). The course number must be specified.

Box A

```
INNER JOIN Course c1 ON c1.calendarID = student.calendarID AND c1.number  = '1300'
LEFT OUTER JOIN Request r1 ON r1.personID = student.personID AND r1.courseID=c1.courseID
INNER JOIN Course c2 ON c2.calendarID = student.calendarID AND c2.number = '1350'
LEFT OUTER JOIN Request r2 ON r2.personID = student.personID AND r2.courseID=c2.courseID
```

Box B

```
AND ((r1.requestID IS NULL
AND r2.requestID IS NOT NULL)
OR (r2.requestID IS NULL AND r1.requestID IS NOT NULL))
```

Students who did not get the Requested Course

This query returns students who did not get scheduled into a course they requested. The course number must be specified.

Box A

```
JOIN Request r ON r.personID = student.personID And r.calendarID = student.calendarID
JOIN Course crs ON crs.courseID = r.courseID
LEFT JOIN (Roster ros JOIN Section sec ON sec.sectionID = ros.sectionID JOIN Trial trl ON trl.trialID = sec.trialID AND trl
.active = 1) ON ros.personID = r.personID AND sec.courseID = r.courseID
```

Box B

```
AND r.type IN ('R','E') AND ros.personID IS NULL
AND crs.number = '1234'
```

Students who have Less Than X Requested Units

This query returns students who have fewer than the specified number of requested units.

Box A

```
INNER JOIN Request r ON r.personID = student.personID
INNER JOIN Course c ON c.courseID = r.courseID
```

Box B

```
AND r.type IN('R','E') AND r.calendarID = student.calendarID
GROUP by student.personID, student.lastName, student.firstName, student.grade, student.studentNumber
HAVING(SUM(c.terms*c.schedules*c.periods))< 48
```

Students Who Have More Than One Course In a Period

This query returns students who are scheduled into more than one course in a period. This only

works before the calendar starts.

Box A

```
INNER JOIN Roster ro ON ro.personID = student.personID and ro.enddate IS NULL
INNER JOIN Section se ON se.SectionID = ro.sectionID
INNER JOIN Course co ON co.courseID = se.courseID AND co.calendarID = student.calendarID
INNER JOIN SectionPlacement sp ON sp.sectionID = se.sectionID
INNER JOIN Period p ON p.periodID = sp.periodID
INNER JOIN PeriodSchedule ps ON ps.periodScheduleID = p.periodScheduleID
INNER JOIN Trial tr ON tr.trialID = se.trialID AND tr.active = 1
```

Box B

```
GROUP BY student.personID, student.lastName, student.firstName, student.grade, student.studentNumber, p.periodID
,
ps.periodScheduleID, sp.termID
HAVING COUNT(se.sectionID) > 1
```

Students in a Blended Learning Group with no Participation Record

This query returns a list of students who are in a Blended Learning Group on the current day (virtual today), but there is no participation record for the student (e.g., the student did not check in or the teacher did not mark the student as participated).

Box A

```
INNER JOIN Roster ro on ro.personID = student.personID
INNER JOIN Section se on se.sectionID = ro.sectionID
INNER JOIN Trial tr on tr.trialID = se.trialID AND tr.active = 1
INNER JOIN Course c on c.courseID = se.courseID AND c.attendance = 1
INNER JOIN Calendar cal on cal.calendarID = c.calendarID AND cal.calendarID = tr.calendarID
LEFT JOIN BlendedLearningGroupAssignment blga on blga.personID = ro.personID AND (blga.endDate is null or blga.e
ndDate >= getDate())
INNER JOIN Day d on d.calendarID = cal.calendarID AND convert(varchar, d.date, 101) = convert(varchar, getDate(), 1
01)
INNER JOIN blendedLearningGroupRemoteDay blgrd on blgrd.calendarID = d.calendarID AND blgrd.dayID = d.dayID A
ND blgrd.blendedLearningGroupID = blga.blendedLearningGroupID
LEFT JOIN studentParticipation sp on sp.sectionID = se.sectionID AND sp.personID = blga.personID AND convert(varch
ar, sp.date, 101) = convert(varchar, d.date, 101)
```

Box B

```
AND (ro.endDate is null or ro.endDate >= getDate())
GROUP BY student.personID, student.lastName, student.firstName, student.grade, student.studentNumber
HAVING sum(cast(sp.participationConfirmation as int)) = 0 OR count(sp.participationConfirmation) = 0
```

Student Queries for Academic Information

Students with Fewer Credits than Desired for Grade

This query returns students enrolled in a certain grade who have less than a specified number of credits. This example returns all twelfth-graders who have fewer than 20 credits.

Box A

```
INNER JOIN v_TranscriptDetail td ON td.personID = student.personID
```

Box B

```
AND student.grade = '12' GROUP BY student.personID, student.lastname, student.firstname, student.grade, student.s
tudentnumber
HAVING(SUM(td.creditsearned))< 20
```

Students on the A Honor Roll

This query returns students who are considered "A" honor-roll students. This query may require user modifications to fit specific needs.

Box A

```
INNER JOIN TermSchedule ts on ts.structureID = student.structureID
INNER JOIN Term t on t.termScheduleID = ts.termScheduleID and t.seq = 1
INNER JOIN GradingTask k on k.name = 'Quarter Grade'
LEFT OUTER JOIN Term tx on tx.termScheduleID = t.termScheduleID and tx.seq = 1
INNER JOIN GradingScore g on g.calendarID = student.calendarID and g.termID = t.termID and g.taskID = k.taskID an
d (g.score like 'A%')
AND g.personID = student.personID LEFT OUTER JOIN GradingScore gx on gx.calendarID = g.calendarID and gx.termI
D = g.termID and gx.taskID = g.taskID
AND gx.personID = g.personID AND NOT (gx.score like 'A%')
```

Box B

```
AND gx.scoreID IS NULL
```

Students on the AB Honor Roll

This query returns students who are considered "A" or "B" honor-roll students. This query may require user modifications to fit specific needs.

Box A

```
INNER JOIN TermSchedule ts on ts.structureID = student.structureID
INNER JOIN Term t on t.termScheduleID = ts.termScheduleID and t.seq = 1
INNER JOIN GradingTask k on k.name = 'Nine Week'
LEFT OUTER JOIN Term tx  on tx.termScheduleID = t.termScheduleID and tx.seq = 1
INNER JOIN GradingScore g on g.calendarID = student.calendarID and g.termID = t.termID and g.taskID = k.taskID an
d (g.score like 'A%'
OR g.score like 'B%') AND g.personID = student.personID
LEFT OUTER JOIN GradingScore gx on gx.calendarID = g.calendarID and gx.termID = g.termID and gx.taskID =  g.task
ID and gx.personID = g.personID
AND NOT (gx.score like 'A%'
OR gx.score like 'B%')
INNER JOIN GradingScore g3 ON g3.calendarID = g.calendarID AND g3.termID = g.termID AND g3.taskID = g.taskID
AND g3.personID = g.personID AND(g3.score LIKE 'B%')
```

Box B

```
AND gx.scoreID IS NULL
```

Students on the "A/B" Honor Roll Excluding Students who only Received "A" Grades

This query returns students who are considered "A" or "B" honor-roll students, but excludes those who received only A's. This query may require user modifications to fit specific needs.

Box A

```
INNER JOIN TermSchedule ts on ts.structureID = student.structureID INNER JOIN Term t on t.termScheduleID = ts.term
ScheduleID and t.seq = 1 INNER JOIN GradingTask k on k.name = 'Quarter'
LEFT OUTER JOIN Term tx on tx.termScheduleID = t.termScheduleID and tx.seq = 1
INNER JOIN GradingScore g on g.calendarID = student.calendarID
AND g.termID = t.termID and g.taskID = k.taskID
AND (g.score like 'A%' OR g.score like 'B%')
AND g.personID = student.personID
LEFT OUTER JOIN GradingScore gx on gx.calendarID = g.calendarID
AND gx.termID = g.termID
AND gx.taskID = g.taskID
AND gx.personID = g.personID
AND NOT (gx.score like 'A%' OR gx.score like 'B%')
INNER JOIN GradingScore g3 ON g3.calendarID = g.calendarID
AND g3.termID = g.termID
AND g3.taskID = g.taskID AND g3.personID = g.personID
AND(g3.score LIKE 'B%')
```

Box B

```
AND gx.personID IS NULL
```

Students who do not have an Assigned Academic Plan

This query returns students who do not have a Graduation Program or Career and Technical Education Program assigned to them.

Box A

```
LEFT JOIN programparticipation pp ON pp.personid = student.personid CROSS JOIN program p
```

Box B

```
AND p.curriculum <> 1
AND pp.personid IS NULL
```

Student Queries for Behavior Information

Students without Behavior Events

This query returns students who do not have behavior event records in the selected calendar.

Box A

```
LEFT OUTER JOIN v_BehaviorDetail be ON student.personID = be.personID and student.calendarID = be.calendarID
```

Box B

```
AND BE.eventID IS NULL
```

This query returns students who do not have behavior event records in the selected calendar and date range.

Box A

```
LEFT OUTER JOIN v_BehaviorDetail bd ON bd.personID = student.personID
AND bd.calendarID = student.calendarID AND bd.incidentDate
BETWEEN 'MM/DD/YYYY' AND 'MM/DD/YYYY'
```

Box B

```
AND bd.eventID IS NULL
```

Students without Behavior Events and Students with Events where Role is Not Offender or Participant

This query returns students who have no behavior events. It also returns students who have events in which they are not marked as an "offender" or as a "participant."

Box A

```
LEFT OUTER JOIN v_BehaviorDetail bd ON bd.personID = student.personID AND bd.calendarID = student.calendarID
```

Box B

```
AND bd.personID IS NULL OR (bd.role <> 'Offender'
AND bd.role <> 'Participant')
```

Students without Behavior Resolutions

This query returns students involved in behavior events without resolution.

Box A

```
INNER JOIN v_BehaviorDetail bd ON bd.personID = student.personID AND bd.calendarID = student.calendarID
```

Box B

```
AND bd.resolutionID IS NULL
```

Students with no Behavior Resolutions in Date Range

This query retrieves students who have no behavior resolutions within the specified date range.

Box A

```
LEFT OUTER JOIN v_BehaviorDetail bd ON bd.personID = student.personID
AND bd.calendarID = student.calendarID AND bd.resolutionStartDate
BETWEEN 'MM/DD/YYYY' AND 'MM/DD/YYYY'
```

Box B

```
AND bd.resolutionID IS NULL
```

Parent and Guardian Related Queries

Students whose Parents/Guardians do not have Email Addresses

This query returns a list of students whose parents/guardians do not have an email address entered on Census Demographics.

Box A

```
JOIN relatedpair rp ON rp.personid1 = student.personid
AND rp.guardian = 1
JOIN contact c ON c.personid = rp.personid2
```

Box B

```
GROUP BY student.personid, student.studentnumber,lastname, firstname, Grade
HAVING count(c.email)=0
```

Students without Guardians

This query returns students without a guardian relationship assigned.

Box A

```
JOIN RelatedPair rp ON rp.personid1 = student.personID
```

Box B

```
GROUP BY student.personID, student.lastName, student.firstName, student.grade, student.studentNumber
HAVING COUNT(guardian)<1
```

Students with Multiple Guardians

This query returns students who have two or more guardian relationships assigned.

Box A

```
INNER JOIN RelatedPair rp ON rp.personID1 = student.personID
AND rp.guardian = 1
```

Box B

```
GROUP BY student.personID, student.lastName, student.firstName, student.grade, student.studentNumber
HAVING COUNT(rp.personID1) > 2
```

Students whose Parents/Guardians do not have a Portal Account

*This query returns students with at least one marked guardian who does not have a Campus Portal account, regardless of household, guardian sequence, or priority. If a parent/guardian is also a staff member and only has a staff account, they will **not** appear in the results.*

Box A

```
JOIN v_CensusContactSummary ccs ON ccs.PersonID = student.personID
LEFT JOIN UserAccount ua ON ua.personID = ccs.contactpersonid
```

Box B

```
AND ua.userID IS NULL and ccs.guardian = 1 AND student.enddate IS NULL
```

Students whose Parents/Guardians do not have a Portal Account (Staff Inclusive Search)

This query returns students in the selected calendar(s) whose parents/guardians do not have a parent portal account. This includes students of staff members with a Campus Application/Campus

Instruction account who do not have a parent portal account. Once a single guardian under the student's relationships has a portal account, no other guardians display on this list. (Note: Parent/guardian relationship is indicated by marking the Guardian and Portal checkboxes on the Relationships screen.)

Box A

```
JOIN relatedpair rp ON rp.personid1 = student.personid
LEFT JOIN useraccount ua ON ua.personid = rp.personid2 AND ua.homepage IS NOT NULL AND ua.homepage <>
'nav-wrapper/TeacherApp/control-center/home'
```

Box B

```
AND rp.guardian = 1 AND rp.portal = 1
GROUP BY student.personid, student.studentnumber, student.firstname, student.lastname, student.grade
HAVING COUNT(ua.personid) = 0
```

Students who do not have at least one guardian with a portal account

This query returns students whose guardians do not have a Campus Portal account.

Box A

```
JOIN relatedpair rp ON rp.personid1 = student.personid
LEFT JOIN useraccount ua ON ua.personid = rp.personid2
```

Box B

```
AND rp.guardian = 1 AND rp.portal = 1
GROUP BY student.personid, student.studentnumber,student.firstname, student.lastname, student.grade
HAVING COUNT(ua.personid) = 0
```

No guardian has a portal account where guardian is true on the relationship

This query returns students who have guardians, but those guardians do not have portal accounts; this query does not use the portal checkbox.

Box A

```
JOIN relatedpair rp ON rp.personid1 = student.personid
LEFT JOIN useraccount ua ON ua.personid = rp.personid2
```

Box B

```
AND rp.guardian = 1
GROUP BY student.personid, student.studentnumber, student.firstname, student.lastname, student.grade
HAVING COUNT(ua.personid) = 0
```

No guardian has a portal account where guardian is in the household and where guardian and portal are both true on the relationship

This query returns students who have guardians, but those guardians do not have portal accounts, and the relationship record indicates portal.

Box A

```
LEFT JOIN (RelatedPair rp
JOIN HouseholdMember hmG ON hmG.personID = rp.personID2 And hmG.endDate IS NULL
JOIN HouseholdMember hmS ON hmS.personID = rp.personID1 And hmS.householdID = hmG.householdID
JOIN UserAccount ua ON ua.personID = rp.personID2) ON rp.personID1 = student.personID And rp.portal = 1 And rp.g
uardian = 1
```

Box B

```
AND (rp.personID1 IS NULL
```

Miscellaneous Student Queries

Finding Students Not Assigned Fees

This query returns all students who have no fees assigned.

Box A

```
LEFT JOIN feeassignment fa on student.personID = fa.personID
```

Box B

```
AND fa.personID is null
```

Students with a Fee Balance for Any Year

This query returns students who owe fees for any school year.

Box A

```
JOIN v_FeeSummary fs on fs.personID = student.personID
```

Box B

```
AND fs.totalBalance <0
```

Students who have a Free/Reduced Eligibility

This query returns a list of students who are eligible for free or reduced lunch.

Box A

```
INNER JOIN v_POSpatron p ON p.personID = student.personID
INNER JOIN v_POSEligibilityCurrent pe on pe.personID = p.personID
LEFT OUTER JOIN POSEligibility pos ON pos.personID = pe.personID
```

Box B

```
AND pos.eligibility IN ('R', 'F')
```

Students who do NOT have a Free/Reduced Eligibility

This query returns a list of students who do not have free or reduced eligibility.

Box A

```
LEFT OUTER JOIN POSEligibility pe ON pe.personid = student.personid
```

Box B

```
AND pe.eligibility IS NULL
```

Find a Student's Eligibility on a Certain Day

This query returns a student's eligibility on a certain day.

Box A

```
INNER JOIN v_POSpatron p ON p.personID = student.personID
INNER JOIN v_POSEligibilityCurrent pe on pe.personID = p.personID
```

Box B

```
AND p.type = 'student' AND (pe.elig_endDate <= GETDATE() OR pe.elig_endDate IS NULL) ORDER BY student.lastnam
e
```

To pull a specific date, change GETDATE to the desired date with single quotes (i.e., '05-23-2011')

Students without a Transportation Code

This query returns students without a transportation code.

Box A

```
LEFT OUTER JOIN Transportation t ON t.personID = student.personID AND t.calendarID = student.calendarID
```

Box B

```
AND t.personID IS NULL
```

Students with Perfect Attendance

This query returns students with perfect attendance records – that is, students with no attendance marks.

Box A

```
LEFT OUTER JOIN attendance a ON a.personID = student.personID AND a.calendarID = student.calendarID
```

Box B

```
AND a.attendanceID IS NULL
```

Impact Aid Employment

This query returns information on students whose parents are employed at Impact Aid sites.

Box A

```
INNER JOIN ImpactAidEmployment i on i.personID = student.personID
```

Box B

```
AND i.siteID is NOT NULL
AND student.endYear = 2010
```

Students Assigned to a Particular Counselor

This query returns students assigned to a specific counselor. Change the lastName and firstName of the counselor in Box B.

Box A

```
INNER JOIN TeamMember tm ON tm.personID = student.personID
AND tm.module  = 'counseling' AND tm.role = 'counselor' AND (tm.endDate IS NULL OR tm.endDate >= getDate())
INNER JOIN individual i ON i.personID = tm.staffPersonID
```

Box B

```
AND i.lastName = 'Staff' AND i.firstName = 'Dean'
```

Students whose Food Service Eligibility Code does not Match their Economic Indicator

This query returns students whose Food Service Eligibility status does not match the Economic Indicator assigned on their enrollment record.

In this example, Code 0 = Ineligible; 1 = Reduced Eligible; 2 = Free Eligible.

Box A

```
INNER JOIN Enrollment e ON e.enrollmentID = student.enrollmentID
AND (e.endDate IS NULL OR e.endDate >= getDate())
LEFT OUTER JOIN POSEligibility pos ON pos.personID = student.personID AND pos.endYear = student.endYear
```

Box B

```
AND ((e.mealStatus = '0' AND pos.eligibility <> 'S')
OR (e.mealStatus = '1' AND pos.eligibility <> 'R')
OR (e.mealStatus =  '2'
AND pos.eligibility <> 'F')
OR (e.mealStatus IS NULL
AND pos.eligibility IS NOT NULL)
OR (e.mealStatus IS NOT NULL
AND pos.eligibility IS NULL))
```

Students Attending School Outside School Boundaries

This query returns information on students who attend a school outside the school boundary defined by their address. It is only applicable if addresses were set up correctly in the schoolBoundary table.

Box A

```
INNER JOIN School sc ON sc.schoolID = student.schoolID
INNER JOIN HouseholdMember hm on hm.personID = student.personID
AND (hm.secondary IS NULL OR hm.secondary = 0)
AND hm.endDate IS NULL
INNER JOIN HouseholdLocation hl ON hl.householdID = hm.householdID
AND (hl.secondary IS NULL or hl.secondary = 0)
AND hl.endDate IS NULL
INNER JOIN Address a ON a.addressID = hl.addressID
INNER JOIN SchoolBoundary sb ON sb.addressID = a.addressID
AND sb.schoolID <> student.schoolID
```

Box B

No text necessary