

LibraryGo query used by Saint Paul Public Schools

Example data:

<b>UniqueID</b> 220916006XXXXX	<b>Student Number</b> 6XXXXX	<b>Name</b> Student, Example	<b>Address</b> 1930 Como Ave {dollar}Saint Paul MN 55108			<b>Homephone</b> 651-555-1234
<b>email</b> <a href="mailto:estudent001@stpaul.k12.mn.us">estudent001@stpaul.k12.mn.us</a>	<b>Birthdate</b> 12/10/2005	<b>pin</b> 12102005	<b>School</b> Crossroads Science	<b>grade</b> 5	<b>status</b> E	

[Type text]

Explanations following query:

1	SELECT DISTINCT '22091600' + s.studentNumber AS UniqueID, s.studentNumber, s.lastName + ', ' + s.firstName AS Name,
2	CASE WHEN pa.postofficebox = '1' THEN LTRIM('PO Box ' + pa.number + '' + COALESCE (pa.street, '') + '' + COALESCE (pa.tag, '') + '' + COALESCE (pa.dir, '') + '' + COALESCE (pa.aprt, '') + '{dollar}' + LTRIM(pa.city1 + ' ' + pa.state1 + ' ' + pa.zip1)) ELSE LTRIM(pa.number + '' + COALESCE (pa.street, '') + '' + COALESCE (pa.tag, '') + '' + COALESCE (pa.dir, '') + '' + COALESCE (pa.aprt, '') + '{dollar}' + LTRIM(pa.city1 + ' ' + pa.state1 + ' ' + pa.zip1)) END AS Address,
3	CASE len(pa.Phone) WHEN 13 THEN replace(replace(pa.phone, '(', ''), ')', '-') WHEN 14 THEN LEFT(replace(replace(pa.phone, '(', ''), ')', '-'), 12) WHEN 15 THEN LEFT(replace(replace(pa.phone, '(', ''), ')', '-'), 12) WHEN 16 THEN LEFT(replace(replace(pa.phone, '(', ''), ')', '-'), 12) WHEN 17 THEN LEFT(replace(replace(pa.phone, '(', ''), ')', '-'), 12) WHEN 18 THEN LEFT(replace(replace(pa.phone, '(', ''), ')', '-'), 12) WHEN 19 THEN LEFT(replace(replace(pa.phone, '(', ''), ')', '-'), 12) WHEN 20 THEN LEFT(replace(replace(pa.phone, '(', ''), ')', '-'), 12) WHEN 21 THEN LEFT(replace(replace(pa.phone, '(', ''), ')', '-'), 12) ELSE " END AS homephone,
4	c.email,
5	CONVERT(varchar, s.birthdate, 110) AS Birthdate,
6	REPLACE(CONVERT(varchar, s.birthdate, 110), '-', '') AS pin,
7	sch.name AS School, s.grade,
8	'E' AS Status, CHAR(10) + CHAR(13) AS CReturn
9	FROM dbo.view_spps_student AS s INNER JOIN dbo.Contact AS c ON c.personID = s.personID AND s.serviceType = 'p'
10	INNER JOIN dbo.view_SPPS_PrimaryAddress AS pa ON s.personID = pa.personid INNER JOIN dbo.School AS sch ON sch.schoolID = s.schoolID AND (sch.number < '826' OR sch.number = '970') AND LEFT(sch.number, 2) <> '17' AND sch.number NOT BETWEEN '701' AND '709' AND sch.number NOT BETWEEN '735' AND '799' AND sch.number NOT IN ('014', '017', '019', '035', '600', '699')
11	INNER JOIN dbo.CustomSchool AS cs ON cs.schoolID = s.schoolID AND cs.attributeID = XXX AND (cs.value = 'P' OR sch.number = '970')
12	WHERE (s.startDate <= GETDATE()) AND (s.personID NOT IN (SELECT personID FROM dbo.CustomStudent WHERE (attributeID = XXXX) AND (value = 1))) AND (s.endDate IS NULL)

[Type text]

2. Case statement to accommodate the few addresses that have a PO Box number. {dollar} added to aid SPPL import.
  3. Case statement removes the parentheses from the area code and handles those numbers that include long distance prefixes.
  5. Convert birthdates to be sure they have slashes not dashes.
  6. Remove slashes and dashes from birthdate.
  8. Only active students are pulled in the query so all status are E for 'enrolled'. CHAR(10) + CHAR(13) AS CReturn An ASCII carriage return is added to enable proper SPPL processing.
  9. s.serviceType = 'p' brings in only primary enrollments so will be one per student with exceptions for user errors (giving a student more than one primary enrollment) which are cleaned up as found.
  10. School numbers are included or excluded to get the full population without dupes.
  11. Cs.value = 'p' includes public schools only; school number XXX includes home schooled.
  12. S.startdate <= GETDATE() excludes future enrollments
- Subquery: excludes students when the parent has chosen to opt out of library access  
s.endDate is null excludes students not currently enrollment.

School numbers, views and attribute values are unique to SPPS. A district using Infinite Campus might have the same field names but values will be different. A district not using Infinite Campus may find the query useful for writing their own query, especially for knowing how to format the data to export. Some values substituted for X to protect privacy.