```
/**
 This example shows the usage of names between the years 1880-2012.

 We represent the data using a filled line plot with time on the
 x-axis and the number of babies named the name on the y-axis.

 This provides an example of picking selective data out of the table.

 C. Andrews
 2014-01-21
 **/

// These are out here to make it easier to change the code to look
// for other names
String targetName = "Christopher";
String targetGender = "M";
Table table;

void setup() {
  size(800, 800);
  // load the table with data
  table = loadTable("names.csv", "header");

  // turn off animation
  noLoop();
}


/**
 This is where all of the hard work is done.
 We iterate through the table looking for rows that match our selected
 name and then mapping those to points in the canvas.

 We are using beginShape() endShape() to create a filled polygon for
 our final visualziation.
 **/
void draw() {
  background(255);
  fill(0, 0, 255);
  // Start the shape and make an initial anchor point in the lower
  // left corner
  beginShape();
  vertex(0, height);

  // the findRows() method returns a list of all rows that match
  // the provided field data (in this case, the value stored in targetName)
  for (TableRow row: table.findRows(targetName,"Name")) {
    int count = row.getInt("Count");
    int year = row.getInt("Year");

    String gender = row.getString("Gender");

    // we are only counting rows that match the same gender
    // we could sum the counts, but we would have to do more work to
    // store the values between rows
    if (gender.equals(targetGender)) {
      // map the count and year to a point and add it to the shape
      float x = map(year, 1880, 2012, 0, width);
      float y = map(count, 0, 60000, height, 0);
      vertex(x, y);
    }
  }
  // finish with a final anchor point in the lower right corner
  vertex(width, height);

  // close the shape
  endShape();
}
```