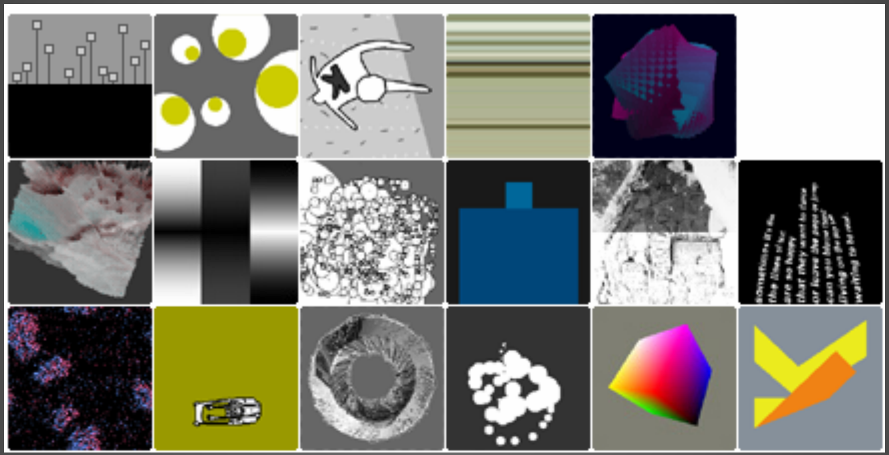
Processing

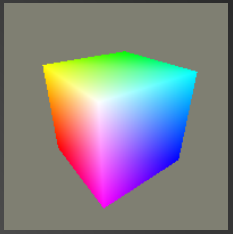


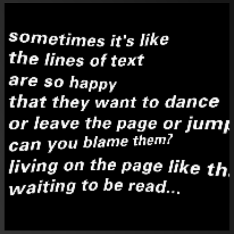
An Introduction

Lonni Besançon

2015

Processing





sometimes it's like
the lines of text
are so happy
that they want to dance
or leave the page or jump
can you blame them?
living on the page like th
waiting to be read...

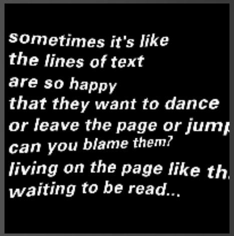
What is it?

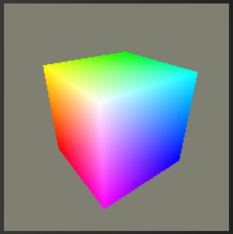
1. a web site
2. a programming environment for learning computational design.
3. a sketchbook for rapidly prototyping
4. 2D/3D graphics api & rendering engine for java
5. open project (by Casey Reas and Ben Fry)
6. an active community of a few thousand people

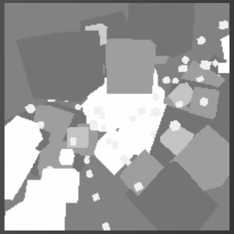
Source: http://acg.media.mit.edu/people/fry/

Processing

- designed to generate and modify images
  - vector/raster drawing,
  - image processing,
  - color models,
  - mouse and keyboard events,
  - network communication,
  - object-oriented programming,
  - Additional libraries...


sometimes it's like
the lines of text
are so happy
that they want to dance
or leave the page or jump
can you blame them?
living on the page like th
waiting to be read...



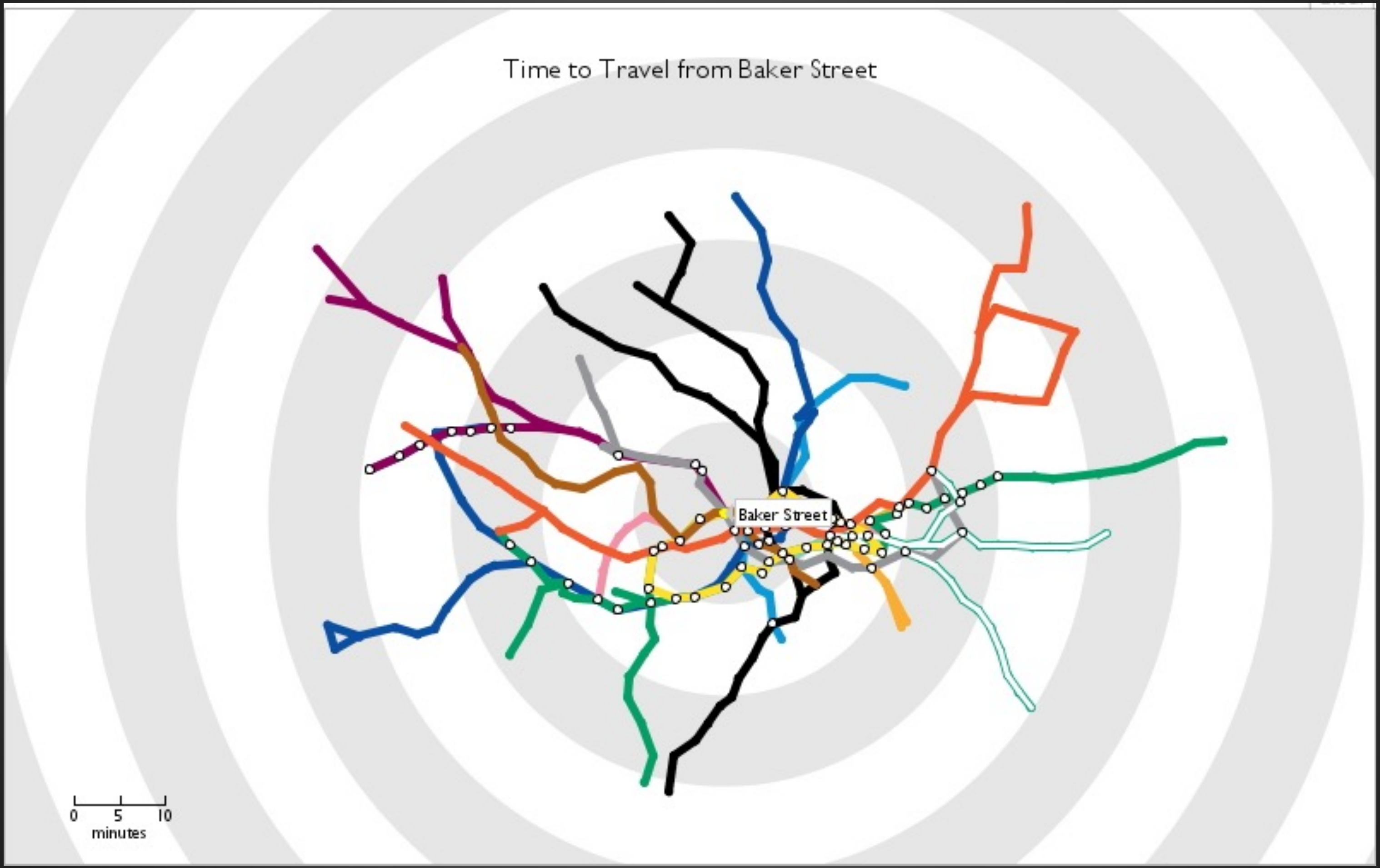


www.processing.org

Examples

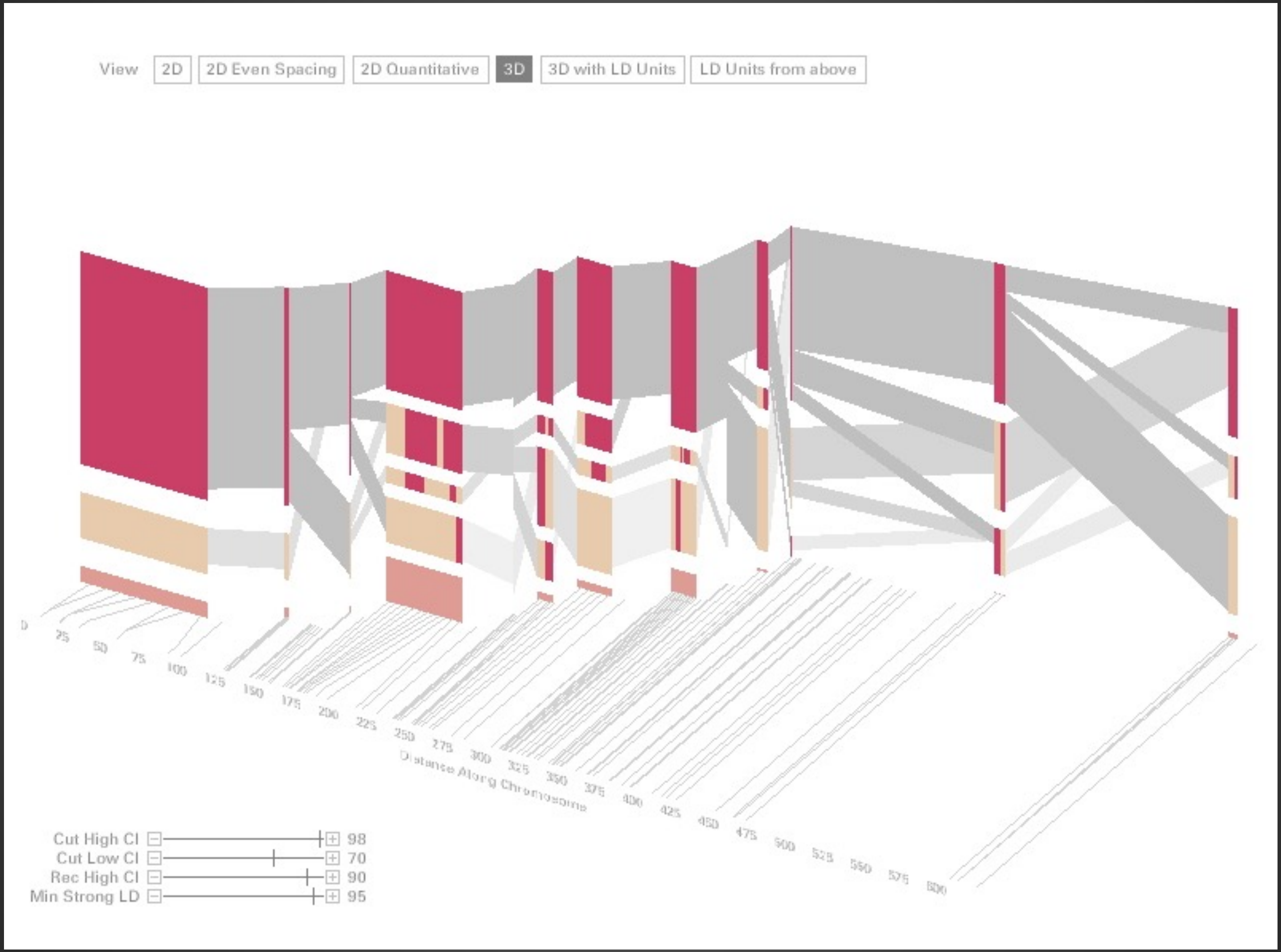
Similar Diversity
Philipp Steinweber & Andreas Koller
(http://similardiversity.net/)

Examples



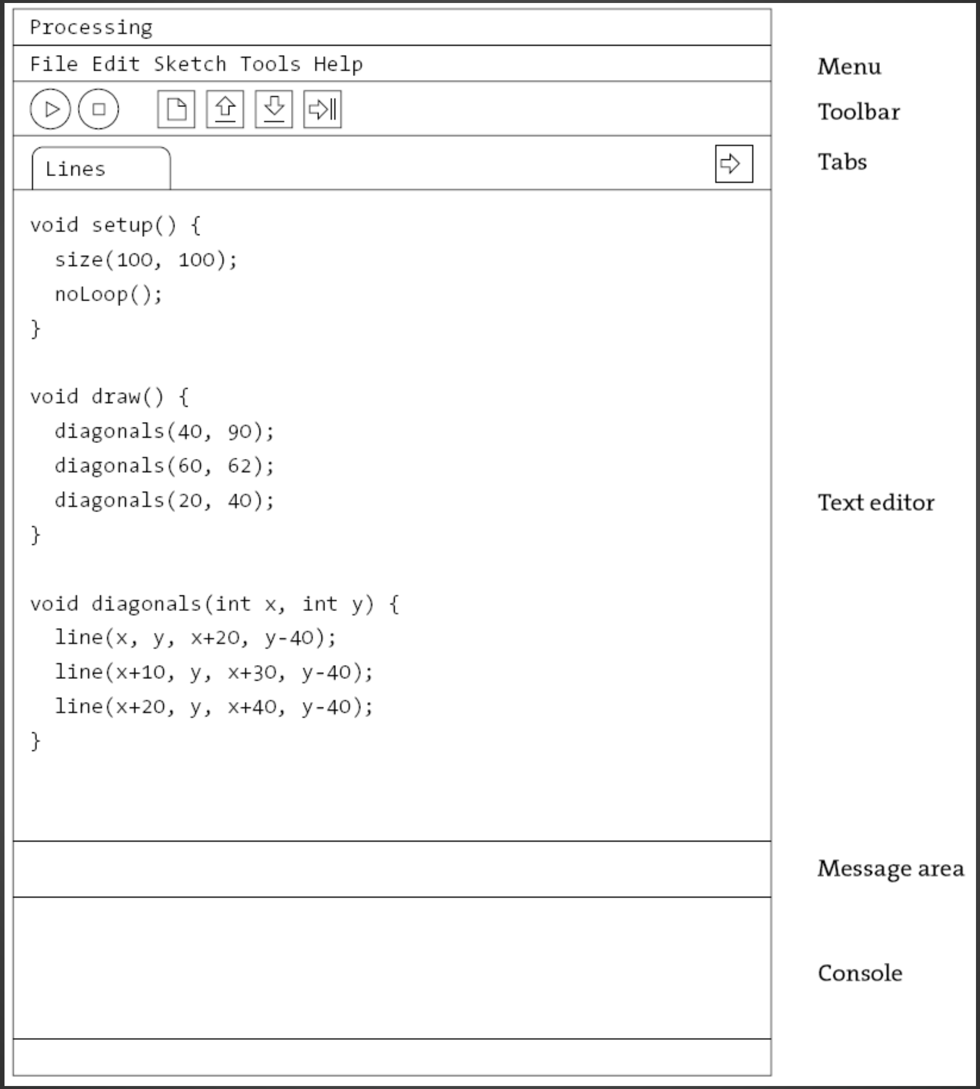
Travel Time Tube Map
Tom Carden

Examples



Visualizing Haplotype
Ben Fry
(Nature Cover)

Getting Started



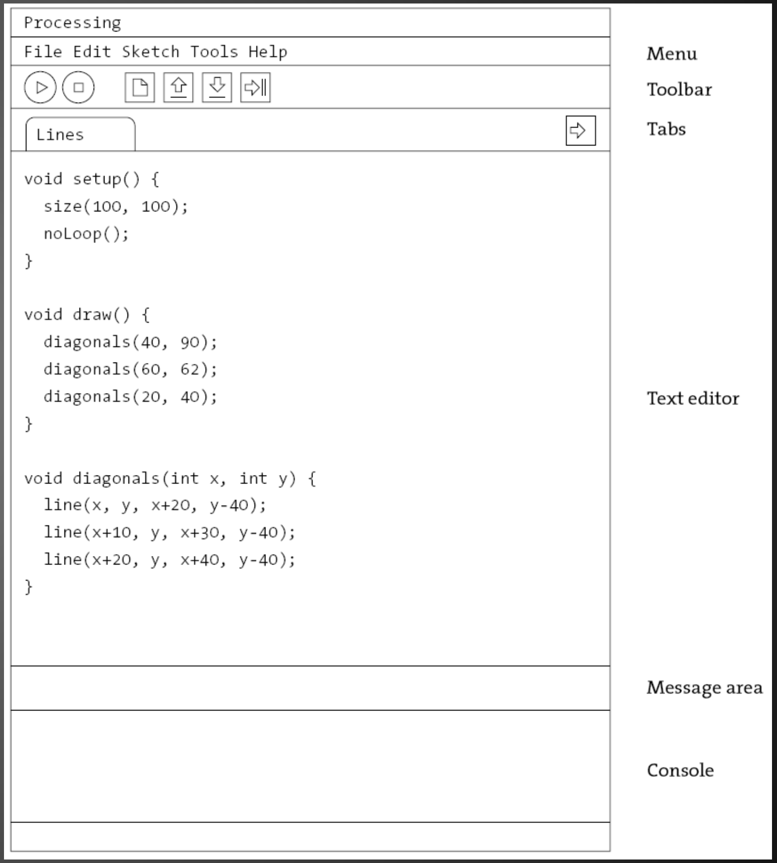
```
void setup() {
  size(100, 100);
  noLoop();
}

void draw() {
  diagonals(40, 90);
  diagonals(60, 62);
  diagonals(20, 40);
}

void diagonals(int x, int y) {
  line(x, y, x+20, y-40);
  line(x+10, y, x+30, y-40);
  line(x+20, y, x+40, y-40);
}
```

Processing Development Environment

[Pictures taken from: Processing, A Programming Handbook]

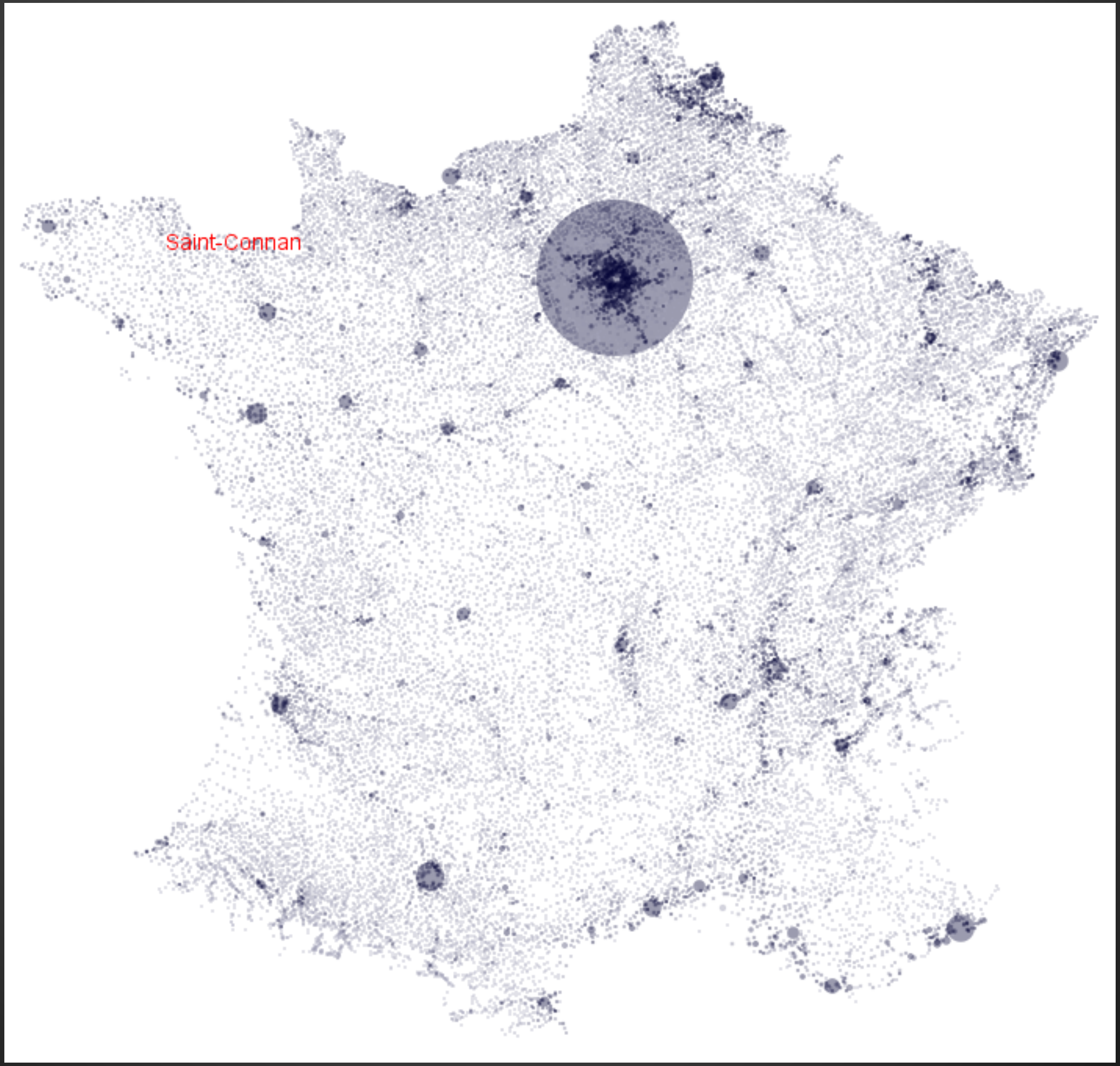
Getting Started

- Sketch = Project
- Can contain several files
  - .pde or .java
- Export → applets



```
void setup() {
  size(100, 100);
  noLoop();
}

void draw() {
  diagonals(40, 90);
  diagonals(60, 62);
  diagonals(20, 40);
}

void diagonals(int x, int y) {
  line(x, y, x+20, y-40);
  line(x+10, y, x+30, y-40);
  line(x+20, y, x+40, y-40);
}
```

Assignment 1



Mapping French towns

- Dataset from this source:
  - http://www.galichon.com/codesgeo/
- Original data:
  - (name, postal code, insee code, long, lat)
  - (name, insee code, population, density)
- Pre-processed for you into
  - (*Postal Code, x, y, inseecode, place, population, surface, altitude*)

Setting up our drawing space

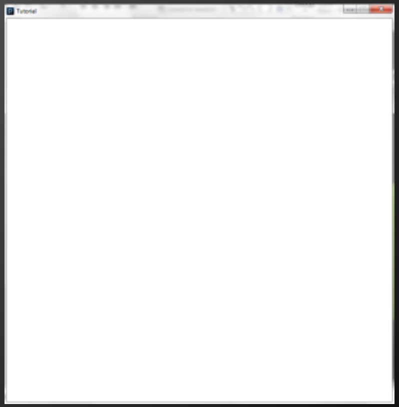
1. Open Processing, on your laptop or:

```
~user/processing-2.0a4/processing
```

2. Write the following code:

```
void setup(){
  size(800,800);
}

void draw(){
  background(255);
}
```

3. Press Run:



Loading the data

- Download and open the following file in a text editor:

http://www.aviz.fr/wiki/uploads/Teaching2015/villes.tsv

Loading the data

- Add the following code to readData
  - Add villes.tsv to your processing project, then add:

```
void readData() {
  String[] lines = loadStrings(“villes.tsv");
  println(lines); // for debug
}
```

  - Call readData() from setup()
  - Press Run

Getting data properties

- Add the following code from:
  - http://www.aviz.fr/Teaching2015/Assignment1

```
//globally
//declare the min and max variables that you need //
in parseInfo
float minX, maxX,float minY, maxY;
int totalCount; // total number of places
int minPopulation, maxPopulation;
int minSurface, maxSurface;
int minAltitude, maxAltitude;

//in your readData method
  String[] lines = loadStrings("villes.tsv");
  parseInfo(lines[0]); // read the header line
```

```
void parseInfo(String line) {
// remove the #
  String infoString = line.substring(2);
  String[] infoPieces = split(infoString, ',');
  totalCount = int(infoPieces[0]);
  minX = float(infoPieces[1]);
  maxX = float(infoPieces[2]);
  minY = float(infoPieces[3]);
  maxY = float(infoPieces[4]);
  minPopulation = float(infoPieces[5]);
  maxPopulation = float(infoPieces[6]);
  minSurface = float(infoPieces[7]);
  maxSurface = float(infoPieces[8]);
  minAltitude = float(infoPieces[9]);
  maxAltitude = float(infoPieces[10]);
}
```

  - Call readData() from setup()
  - Press Run

Getting data properties

- readData() should look like that

```
void readData() {
  String[] lines = loadStrings(“villes.tsv");
  // println(lines); // for debug
  parseInfo(lines[0]); // read the header line
}
```

  - Press Run

Reading data

- Add the following code to readData

```
float x[]; float y[];

void readData() {
…
  x = new float[totalCount];
  y = new float [totalCount];
  for(int i = 2; i < totalCount; ++i){
    String pieces[] = split(lines[i], TAB);
    x[i-2] = float (pieces[1]);
    y[i-2] = float (pieces[2]);
  }
}
```

  – Press Run

Finally – drawing!

- Add the following code to draw()

```
background(255);
   color black = color(0);
   for(int i = 0; i < totalCount; ++i){
      set((int)x[i],(int)y[i],black);
   }
```

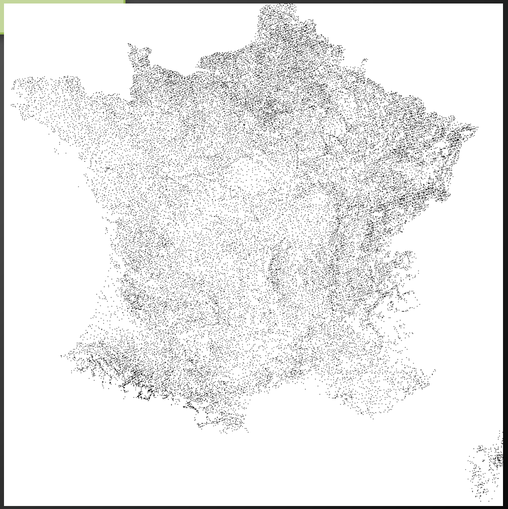
  - Press Run

Finally – drawing!

- Fixing the drawing

```
float mapX(float x) {
   return map(x, minX, maxX, 0,800);
 }
 float mapY(float y) {
   return map(y, minY, maxY, 800,0);
 }
```

  - use in set( ) call
  - Press Run

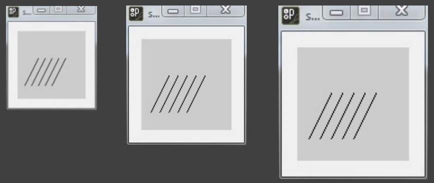


How to draw each town?

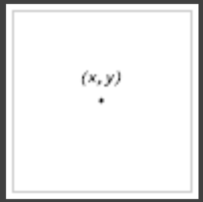
- Open Menu Help -> Reference

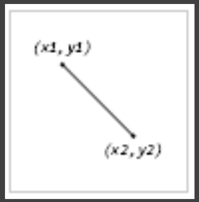
Coordinates and Primitives

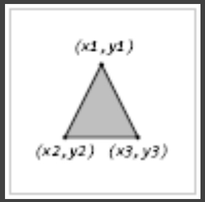
size(),

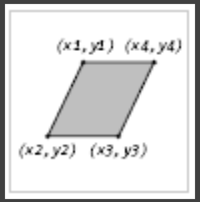


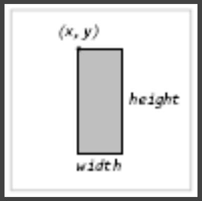
point(), line(), triangle(), quad(), rect(), ellipse(), bezier()

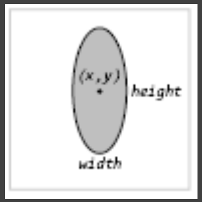


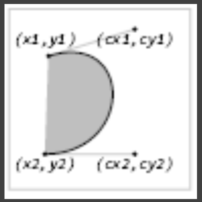




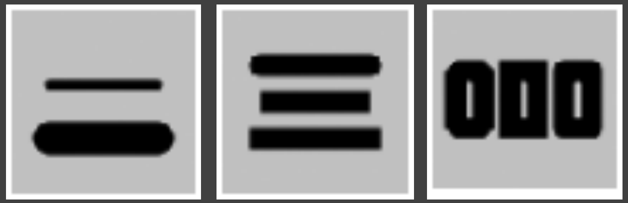




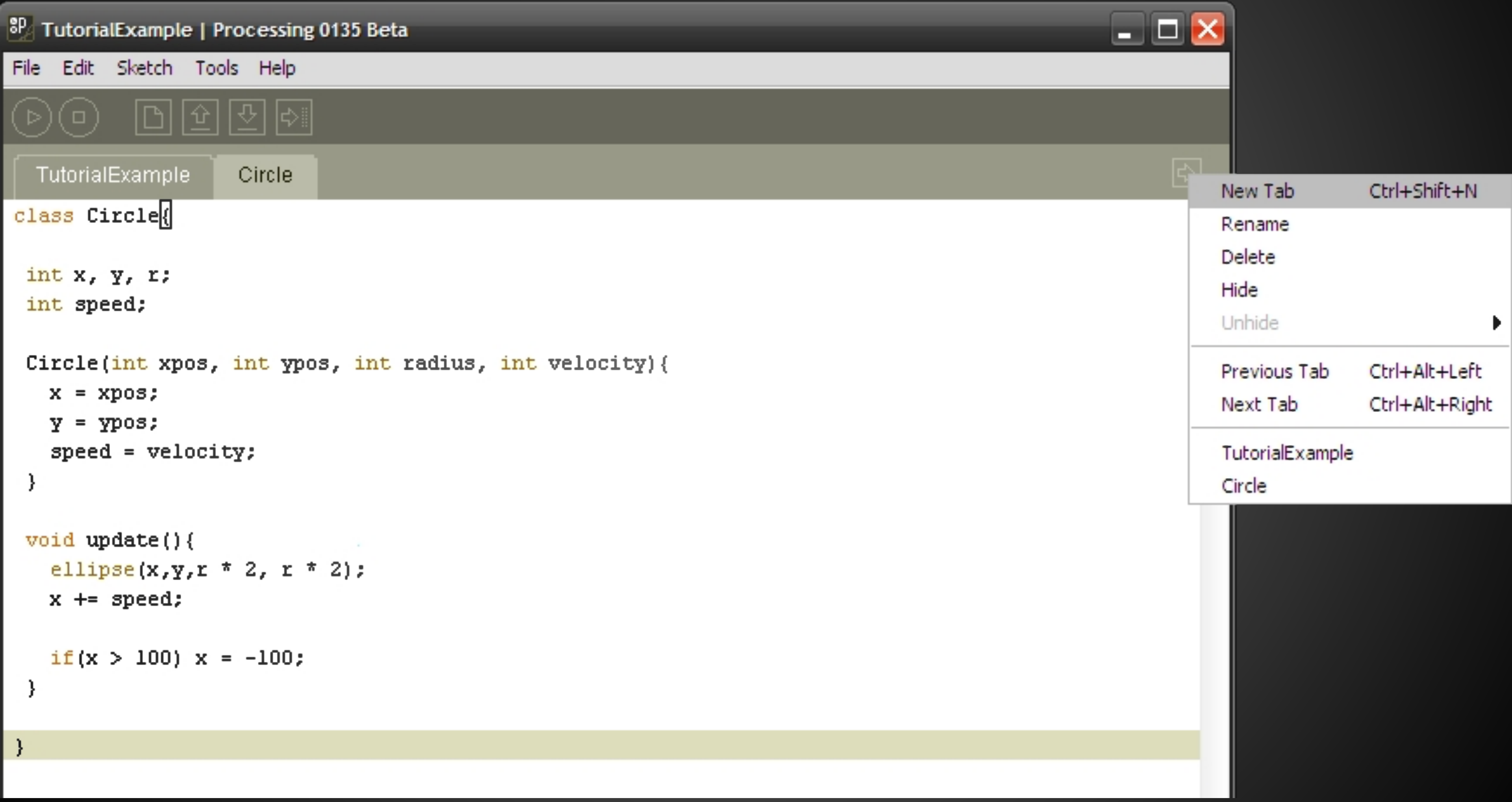




background(), fill(), stroke(), noFill(), noStroke(), strokeWeight(), strokeCap(), strokeJoin(), smooth(), noSmooth(), ellipseMode(), rectMode()



Create the class “Place”

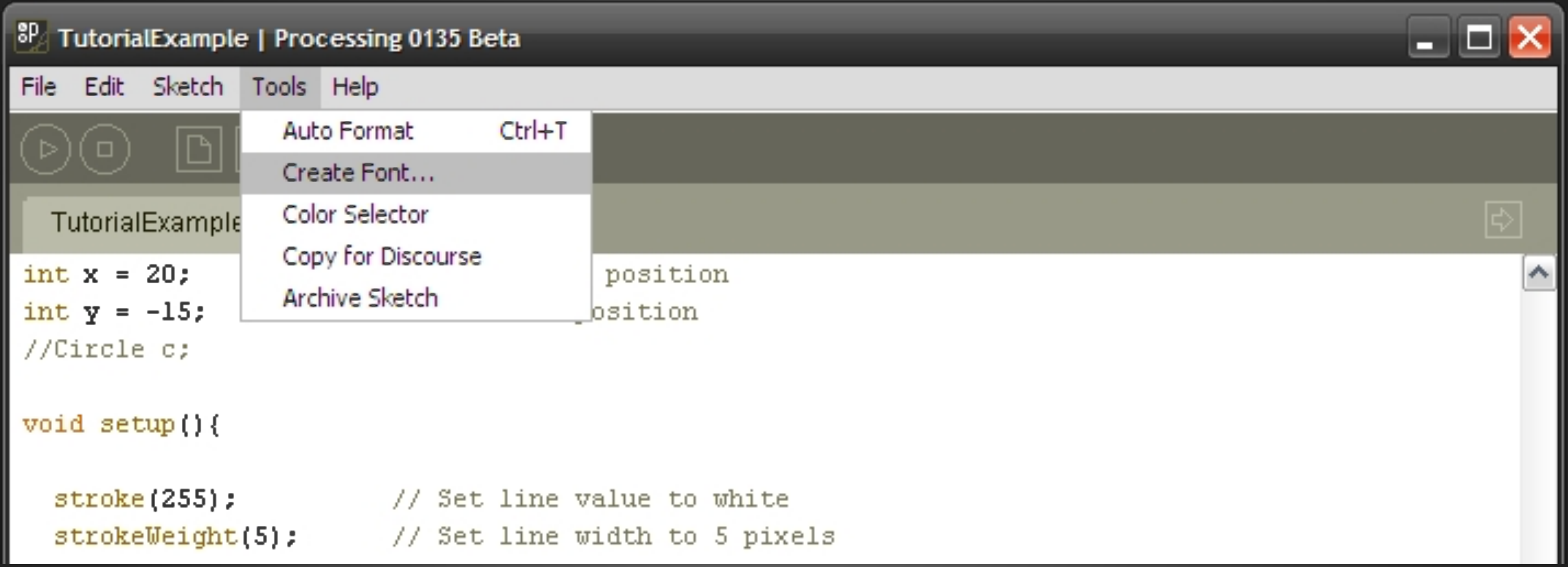


```
class Circle{

 int x, y, r;
 int speed;

 Circle(int xpos, int ypos, int radius, int velocity){
   x = xpos;
   y = ypos;
   speed = velocity;
 }

 void update(){
   ellipse(x,y,r * 2, r * 2);
   x += speed;

   if(x > 100) x = -100;
 }

}
```

Create the class “Place”

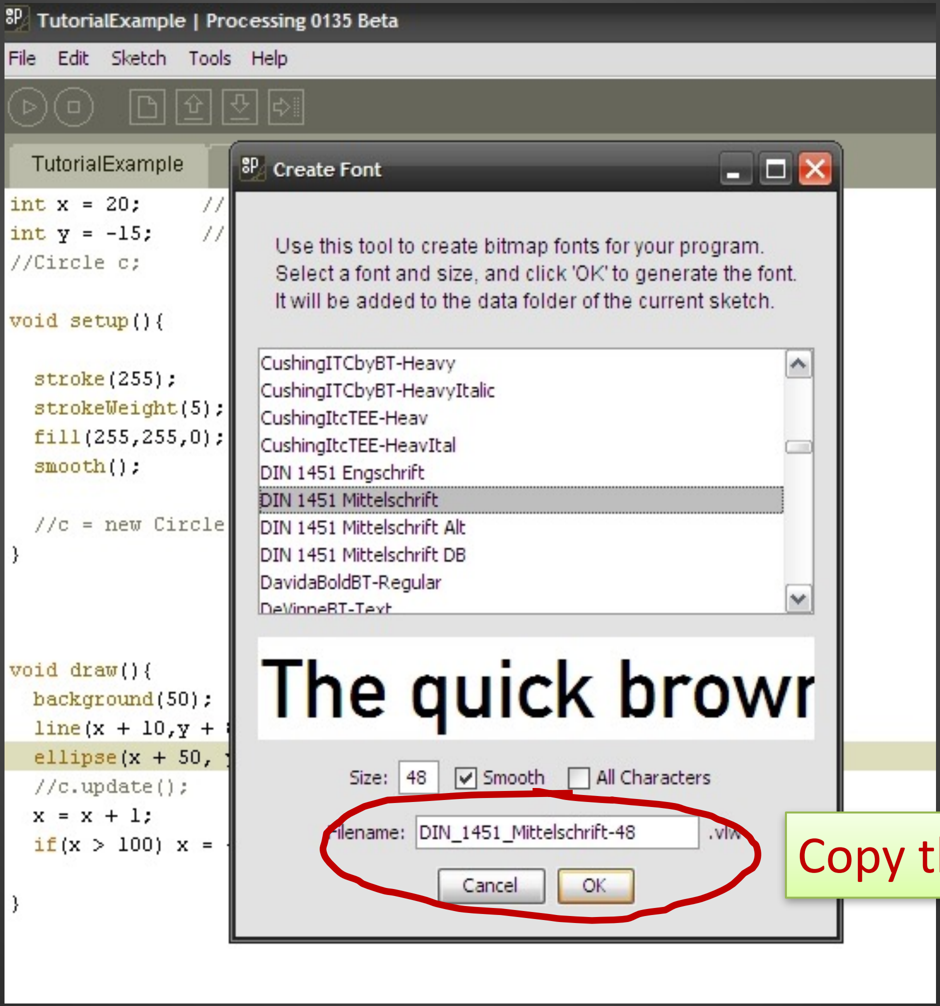
```
class Place {
  int postalcode;
  String name;
  float x;
  float y;
  float population;
  float density;


…

  put a drawing function in here and call from
main drawing loop

}
```

Text Rendering



```
int x = 20;                              position
int y = -15;                            osition
//Circle c;

void setup(){

  stroke(255);          // Set line value to white
  strokeWeight(5);      // Set line width to 5 pixels
```

Text Rendering


TutorialExample | Processing 0135 Beta
File Edit Sketch Tools Help
TutorialExample
Create Font

```
int x = 20;       //
int y = -15;      //
//Circle c;

void setup(){

  stroke(255);
  strokeWeight(5);
  fill(255,255,0);
  smooth();

  //c = new Circle
}



void draw(){
  background(50);
  line(x + 10,y +
  ellipse(x + 50,
  //c.update();
  x = x + 1;
  if(x > 100) x =

}
```

Use this tool to create bitmap fonts for your program. Select a font and size, and click 'OK' to generate the font. It will be added to the data folder of the current sketch.

- CushingITCbyBT-Heavy
- CushingITCbyBT-HeavyItalic
- CushingItcTEE-Heav
- CushingItcTEE-HeavItal
- DIN 1451 Engschrift
- DIN 1451 Mittelschrift
- DIN 1451 Mittelschrift Alt
- DIN 1451 Mittelschrift DB
- DavidaBoldBT-Regular
- DeVinneBT-Text

The quick browr

Size: 48 [✓ Smooth] All Characters

Filename: DIN_1451_Mittelschrift-48 .vlw

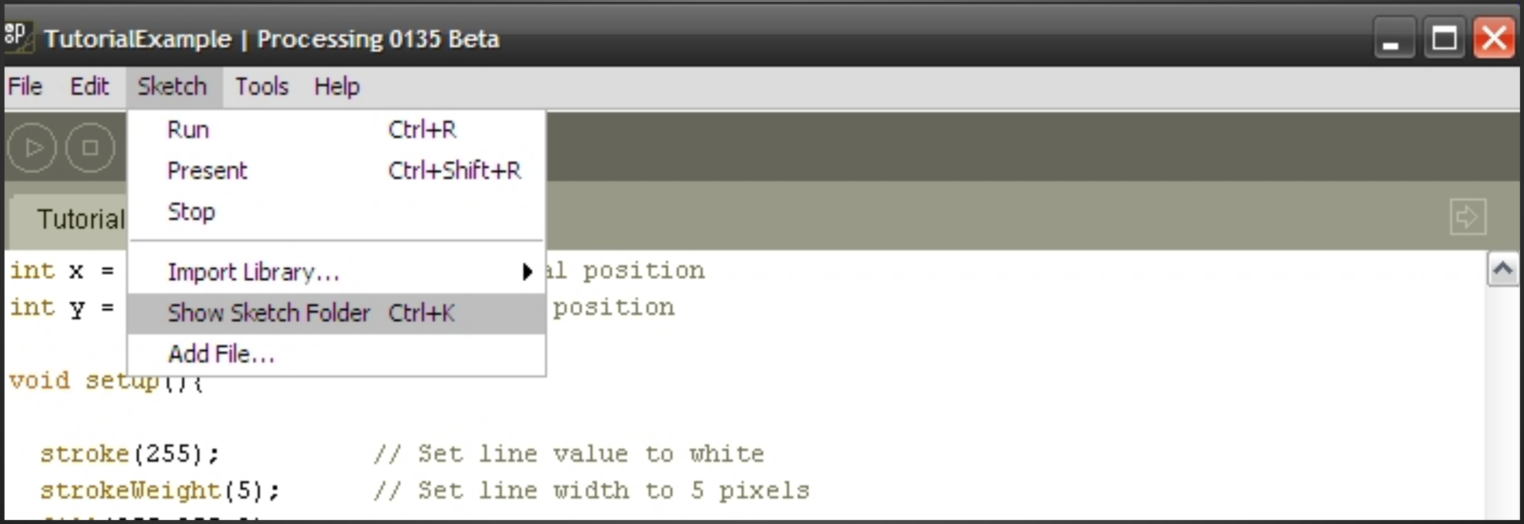
Cancel OK

Copy the Name!

Text Rendering

Double check:

- font file should be in data folder



```
int x =                    al position
int y =                     position

void setup(){

  stroke(255);          // Set line value to white
  strokeWeight(5);      // Set line width to 5 pixels
```

Text Rendering

```
PFont font;   //initialize font variable

void setup(){
  [..]
  font = loadFont("Humanist521BT-Roman-48.vlw");
  textFont(font,32);
}

void draw(){
  background(50);   // clears the background in black
  text("move", 10, 50);

  [..]
}
```

Interaction - Mouse

- Position
  - Replace code & Run

```
text("move", mouseX, mouseY);
```

- Buttons

```
if (mousePressed == true) …
if (mouseButton    == left) …
```

- Mouse Functions

```
void mousePressed( )  {..}
void mouseDragged( ) {..}
void mouseReleased() {..}
```

Interaction - Keyboard

- Events

```
if(keyPressed) {
      if(key >= 'A' && key <= 'z') { […] }
}
```

- Functions

```
void keyPressed( ) { […] }
```

Before next lecture

- Finish assignment 1: http://www.aviz.fr/Teaching2015/Assignment
- Look at project info: http://www.aviz.fr/Teaching2015/Projects
- Fill in the google doc here: https://docs.google.com/spreadsheets/d/1clBPOlNT4UQLJuMhFDh887AWy33rubb5r3Cumu4kxoQ/edit#gid=0&vpid=A1
- Fille in this second google doc: https://docs.google.com/spreadsheets/d/1-tq1h1MDw4zF1frs1-HiONxy1x4p5Onmiz5sVBFIYqY/edit#gid=0&vpid=A1
- For three groups: prepare your paper presentation