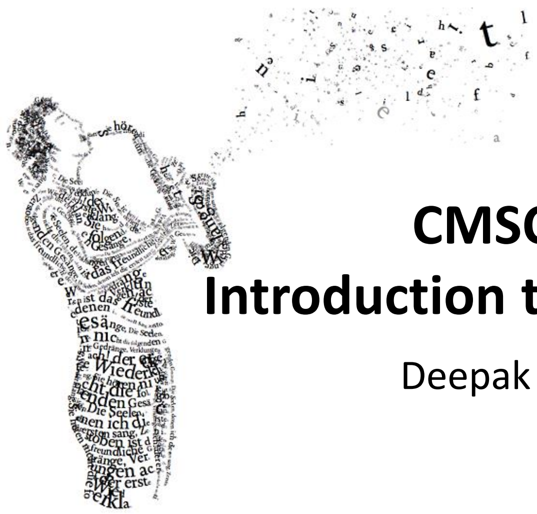
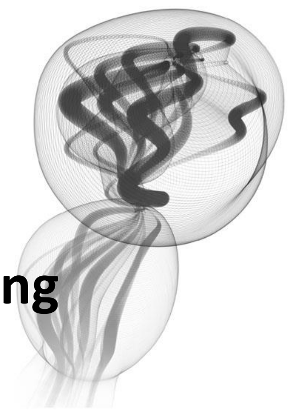




CMSC110 Introduction to Computing

Deepak Kumar

Administrivia

CMSC110: Introduction to Computing

Fall 2019

Course Website: https://cs.brynmawr.edu/Courses/cs110/fall2019/
Instructor:

Deepak Kumar, (dkumar@cs.brynmawr.edu)

Lectures

TuTh 12:55p to 2:15p in Park 245

TA-Support

>20 hrs/week in Park 230/231

Labs – **Register and attend one of these**

- Section A: Tuesdays 2:15 p.m. to 3:15 p.m. (led by Prof. Kumar)
- Thursdays 11:55 a.m. to 12:45 p.m. (led by Prof. Kumar)

Office Hours

Wednesdays 2:00 to 4:00p

Grading

• ~7 Assignments	30%
• Lab Attendance	10%
• Exam 1	20%
• Exam 2	20%
• Exam 3	25%
Total	100%

2

Administrivia

Software

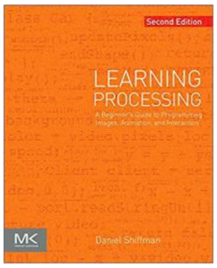
Processing 3.X

- Already installed in the CS Lab
- Also available for your own computer @ www.processing.org
- Processing == Java

Required

Learning Processing: A Beginner's Guide to Programming Images, Animations, and Interaction, 2nd Edition by Daniel Shiffman, Publisher: Morgan Kauffmann, 2015. Available at the Campus Bookstore. Also at amazon for $34.97 (as of August 22, 2-19).

Dropbox Account: Please go to dropbox.com and register. You will be using dropbox to submit many of your assignments. You will need to have this set up by the end of Week#1.



3

Class Lottery

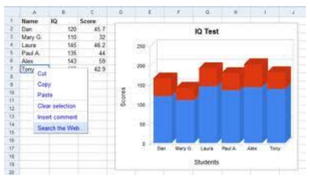
- Make sure to sign-in your name.
- If you are not on the class list, sign on the attached sheet. We will contact you by e-mail as soon as we have confirmation from other students.

4

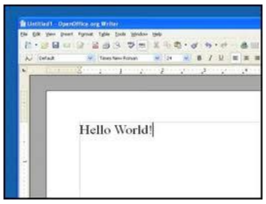
What is Computing?

5

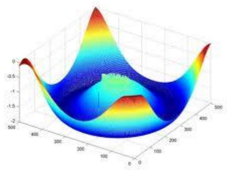
Computing: Your Parent's View

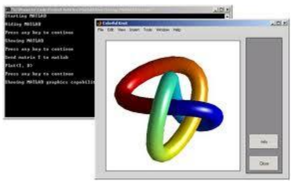


	A	B	C
1	Name	IQ	Score
2	Dan	120	45.7
3	Mary G.	110	32
4	Laura	145	46.2
5	Paul A.	135	44
6	Alex	143	59
7	Tony	[illegible]	42.9



Hello World!





```
Starting MATLAB
Hiding MATLAB
Press any key to continue
Showing MATLAB
Press any key to continue
Send matrix I to matlab
Plot(I, 8)
Press any key to continue
Showing MATLAB graphics capabili
```

6

Computing: internet, e-mail, network…

SURF.....INTERNET.....NETWORK
HTTP:WWW....COM.....WWW.....SURF.....INTERNET...NETWORK

7

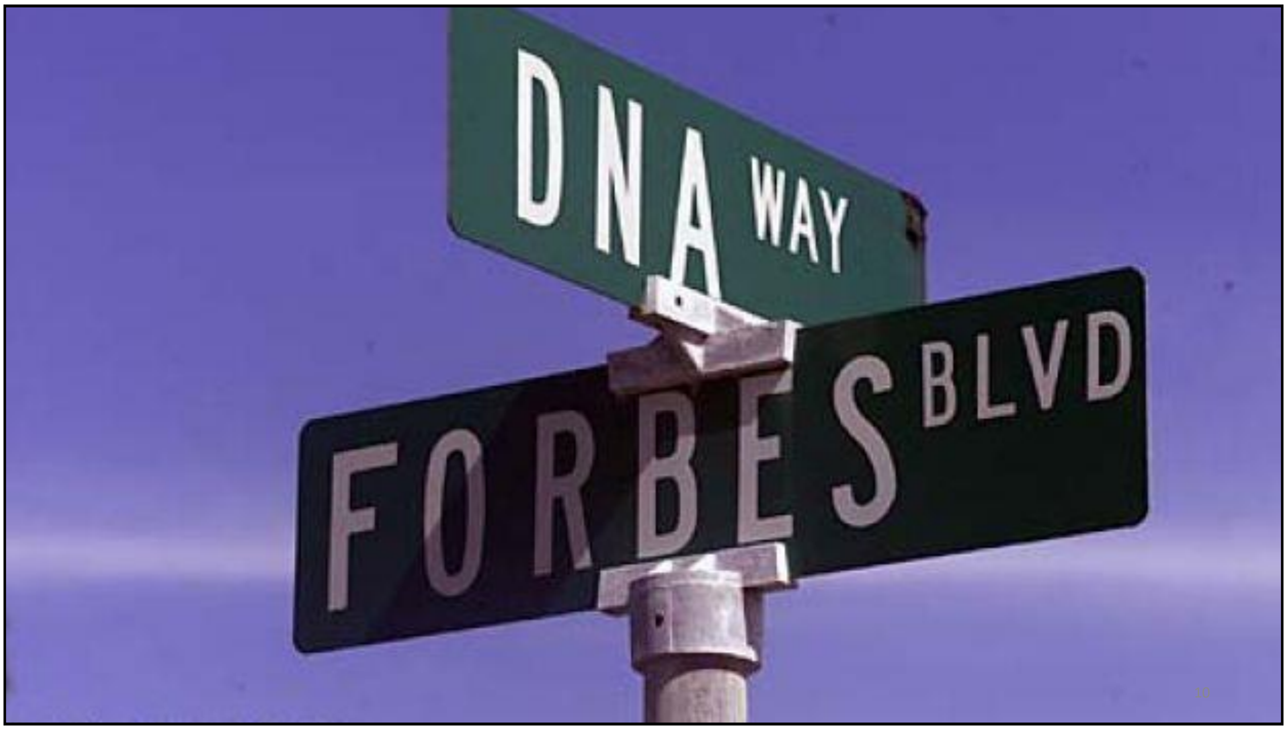
Computing: Digital Photography

http://www.alanzeyes.com/2009/02/hdr-photography.html

8

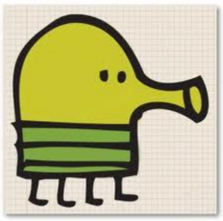
Computing: Entertainment…

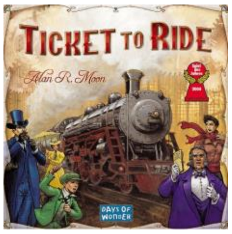
9



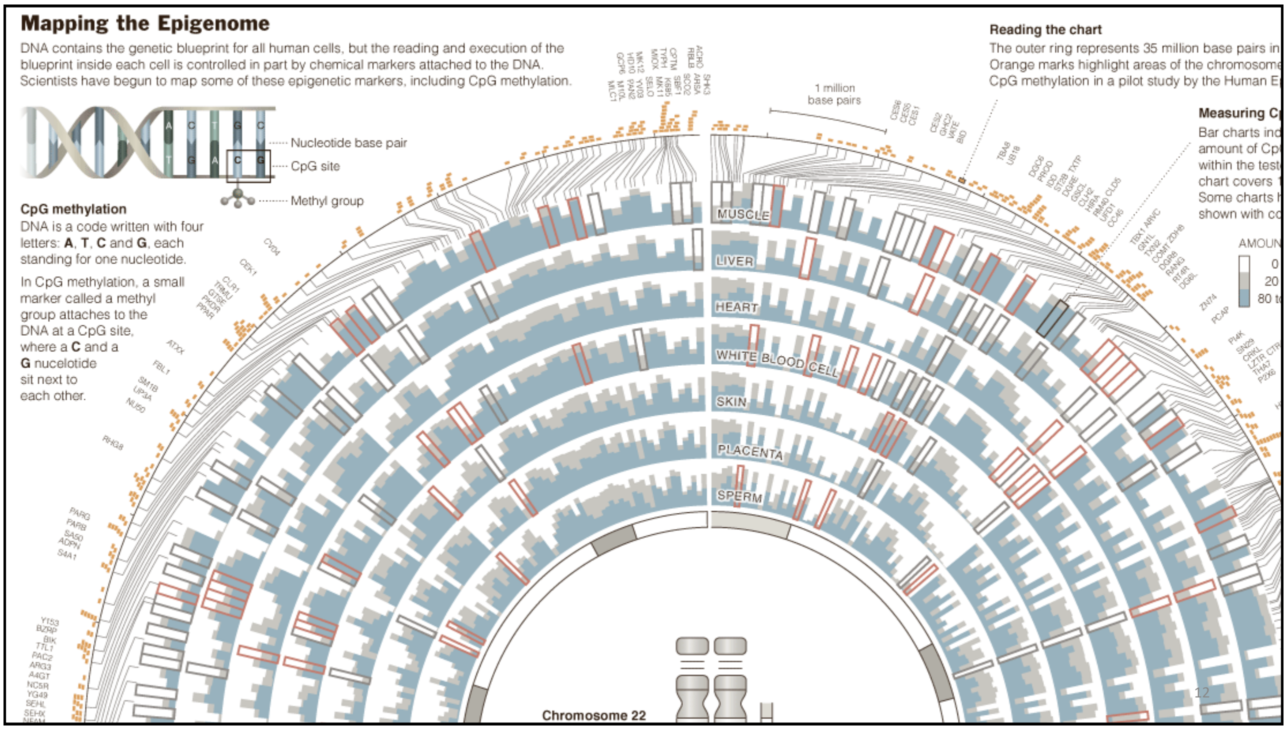
10

Computing: Entertainment…





11



Mapping the Epigenome

DNA contains the genetic blueprint for all human cells, but the reading and execution of the blueprint inside each cell is controlled in part by chemical markers attached to the DNA. Scientists have begun to map some of these epigenetic markers, including CpG methylation.

CpG methylation

DNA is a code written with four letters: **A**, **T**, **C** and **G**, each standing for one nucleotide.

In CpG methylation, a small marker called a methyl group attaches to the DNA at a CpG site, where a **C** and a **G** nucelotide sit next to each other.

Reading the chart

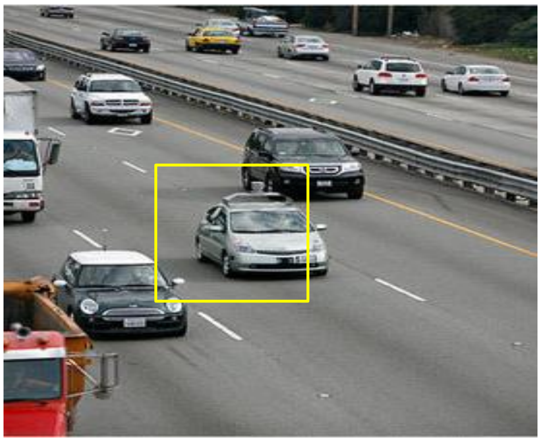
The outer ring represents 35 million base pairs in
Orange marks highlight areas of the chromosome
CpG methylation in a pilot study by the Human E

Measuring C

Bar charts ind
amount of Cp
within the test
chart covers
Some charts
shown with c

12

Self-driving (Autonomous) Cars



13

Some Areas in Computer Science

Artificial Intelligence

Robotics

Human-Computer Interaction

Computer Graphics

Computer Vision

Operating Systems

Computer Networking

Databases

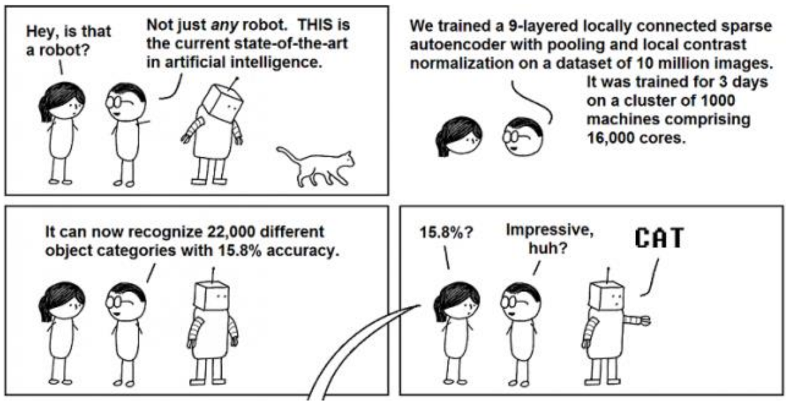
Computer Security

Ubiquitous Computing

14

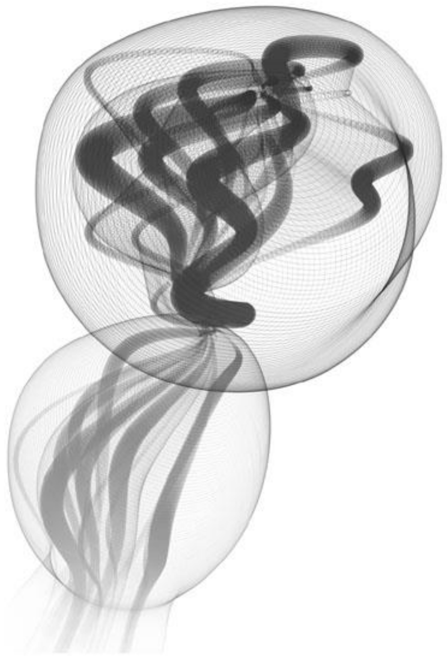
More trendy…

- Machine Learning (Deep Learning)
- Data Science (Big Data)
- Cybersecurity


Hey, is that a robot?
Not just *any* robot. THIS is the current state-of-the-art in artificial intelligence.
We trained a 9-layered locally connected sparse autoencoder with pooling and local contrast normalization on a dataset of 10 million images. It was trained for 3 days on a cluster of 1000 machines comprising 16,000 cores.
It can now recognize 22,000 different object categories with 15.8% accuracy.
15.8%?
Impressive, huh?
CAT

15

ART



Protobytes
By Ira Greenberg

16

What is Computer Science?

Computer science is the study of solving problems using computation

– Computers are part of it, but the emphasis is on the problem solving aspect

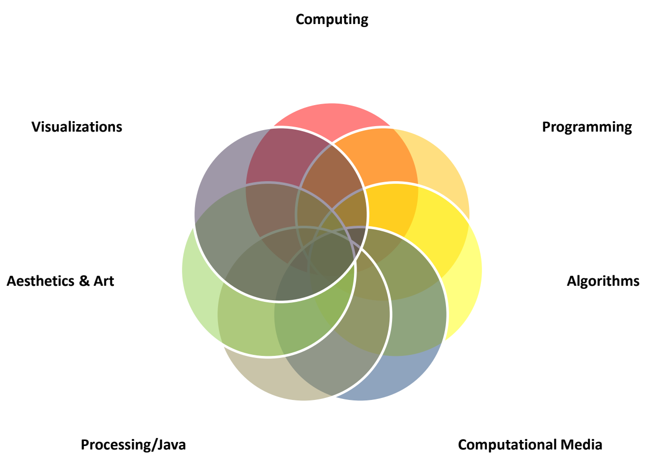
Computer scientists work across disciplines:

Mathematics	Geoscience	Medicine/Surgery
Biology (bioinformatics)	Archaeology	Engineering
Chemistry	Psychology	Linguistics
Physics	Sociology	Art
Geology	Cognitive Science	…

17

Introduction to ^ Computing

Creative



18

Algorithms

An **algorithm** is an effective method for solving a problem expressed as a finite sequence of instructions. For example,

Put on shoes

left sock
right sock
left shoe
right shoe

Programming = Writing Apps

Programming is the process of designing, writing, testing, debugging / troubleshooting, and maintaining the source code of computer programs.

This source code is written in a **programming language**.

A program

```
int areaOfCircle(int radius){
  return PI*radius*radius;
}

r = 10;
area = areaOfCircle(r);
```

21

Programming Languages

Processing/Java/C/C++	Python	Lisp

`int areaOfCircle(int radius){`
`   return PI*radius*radius;`
`}`
 	
`r = 10;`
`area = areaOfCircle(r);`
	`def areaOfCircle(radius):`
	`   return PI*radius*radius;`
	
	`r = 10`
	`area = areaOfCircle(r)`
		`(defun areaOfCircle (radius)`
		`   (return (* PI radius radius)))`
		
		`(setq r 10)`
		`(setq area (areaOfCircle r))`

22

Programming Languages

Processing	Python	Lisp

```
int areaOfCircle(int radius){
   return PI*radius*radius;
}

r = 10;
area = areaOfCircle(r);
```
	```
	def areaOfCircle(radius):
	   return PI*radius*radius;


	r = 10
	area = areaOfCircle(r)
	```
		```
		(defun areaOfCircle (radius)
		   (return (* PI radius radius)))


		(setq r 10)
		(setq area (areaOfCircle r))
		```

FORTRAN, BASIC, Pascal, C, Ada, C++, C#, Java, Javascript, Perl, Ruby, Swift, R…

There are over 3000 of them!

A more interesting program…

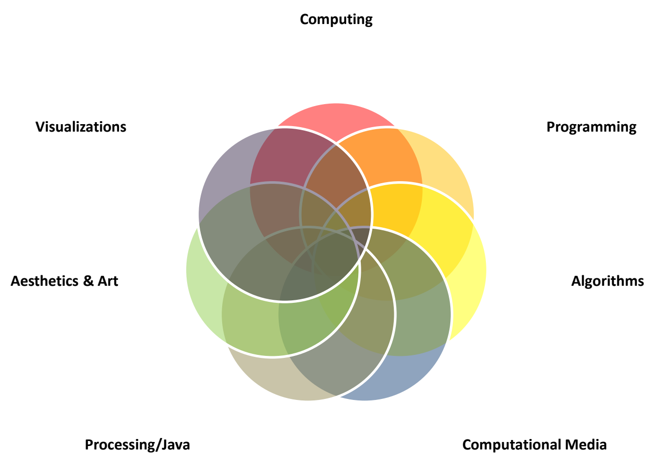
```
Eye e1, e2, e3, e4, e5;

void setup()
{
  size(200, 200);
  smooth();
  noStroke();
  e1 = new Eye( 50,  16,  80);
  e2 = new Eye( 64,  85,  40);
  e3 = new Eye( 90, 200, 120);
  e4 = new Eye(150,  44,  40);
  e5 = new Eye(175, 120,  80);
} // setup()

void draw()
{
  background(102);

  e1.update(mouseX, mouseY);
  e2.update(mouseX, mouseY);
  e3.update(mouseX, mouseY);
  e4.update(mouseX, mouseY);
  e5.update(mouseX, mouseY);

  e1.display();
  e2.display();
  e3.display();
  e4.display();
  e5.display();
} // draw()
```

```
class Eye
{
  int ex, ey;
  int size;
  float angle = 0.0;

  Eye(int x, int y, int s) {
    ex = x;
    ey = y;
    size = s;
  } // Eye()

  void update(int mx, int my) {
    angle = atan2(my-ey, mx-ex);
  } // update()

  void display() {
    pushMatrix();
    translate(ex, ey);
    fill(255);
    ellipse(0, 0, size, size);
    rotate(angle);
    fill(153);
    ellipse(size/4, 0, size/2, size/2);
    popMatrix();
  } // display()
} // class Eye
```

Our Goal

- Use computing to realize works of art
- Explore new metaphors from computing: images, animation, interactivity, visualizations
- Learn the basics of computing
- Have fun doing all of the above!

Introduction to ^ Creative Computing



Let’s get started…

27

Administrivia

Software

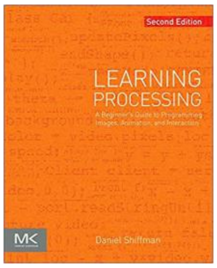
Processing 3.X

- Already installed in the CS Lab
- Also available for your own computer @ www.processing.org
- Processing == Java

Required

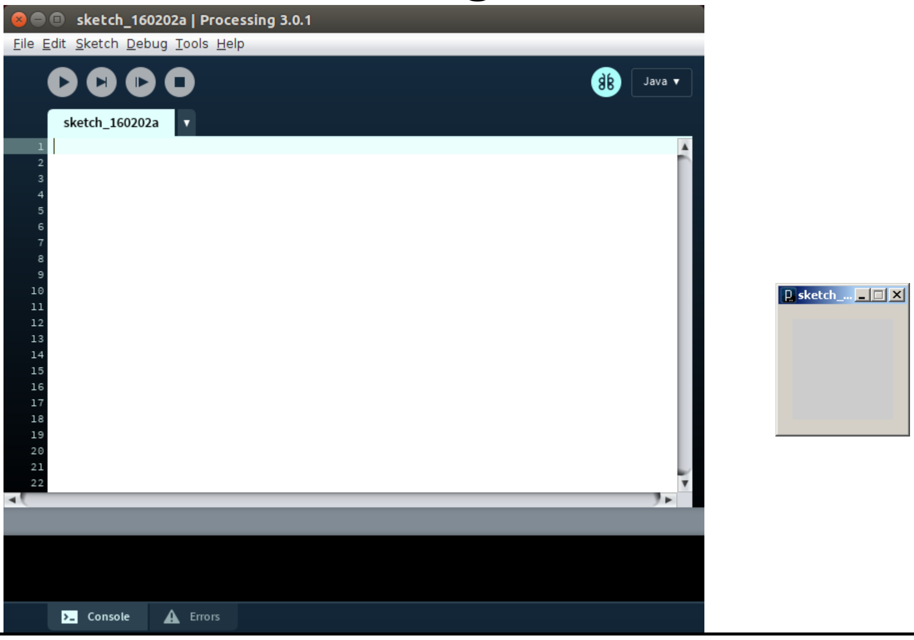
Learning Processing: A Beginner's Guide to Programming Images, Animations, and Interaction, 2nd Edition by Daniel Shiffman, Publisher: Morgan Kauffmann, 2015. Available at the Campus Bookstore. Also at amazon for $34.97 (as of August 22, 2-19).

Dropbox Account: Please go to dropbox.com and register. You will be using dropbox to submit many of your assignments. You will need to have this set up by the end of Week#1.

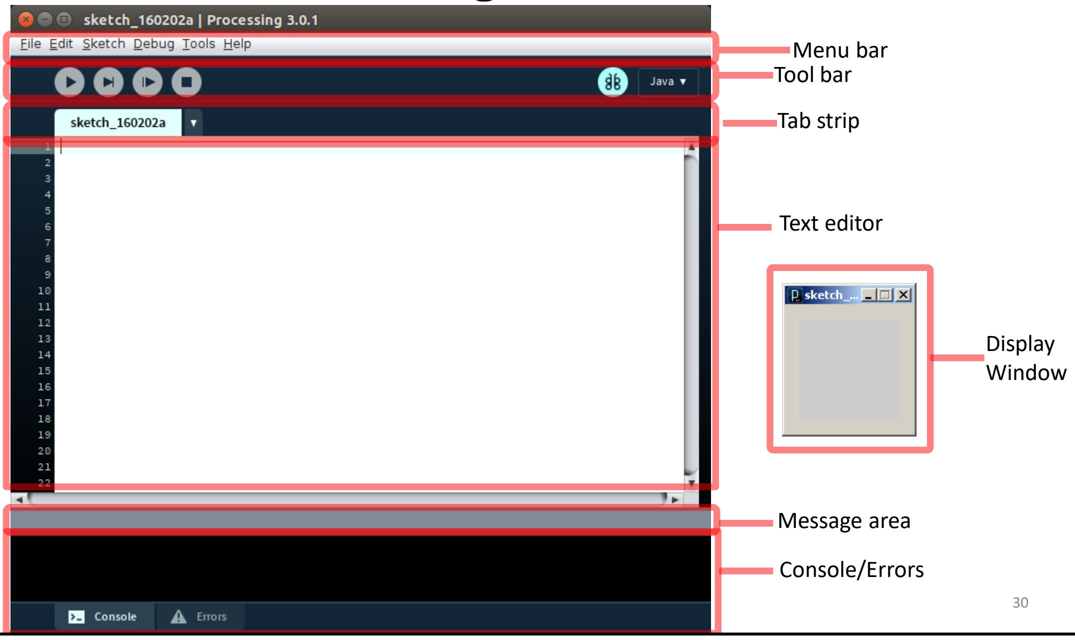


28

Processing 3.0 IDE


sketch_160202a | Processing 3.0.1
File Edit Sketch Debug Tools Help
Java
sketch_160202a
Console
Errors
sketch_...

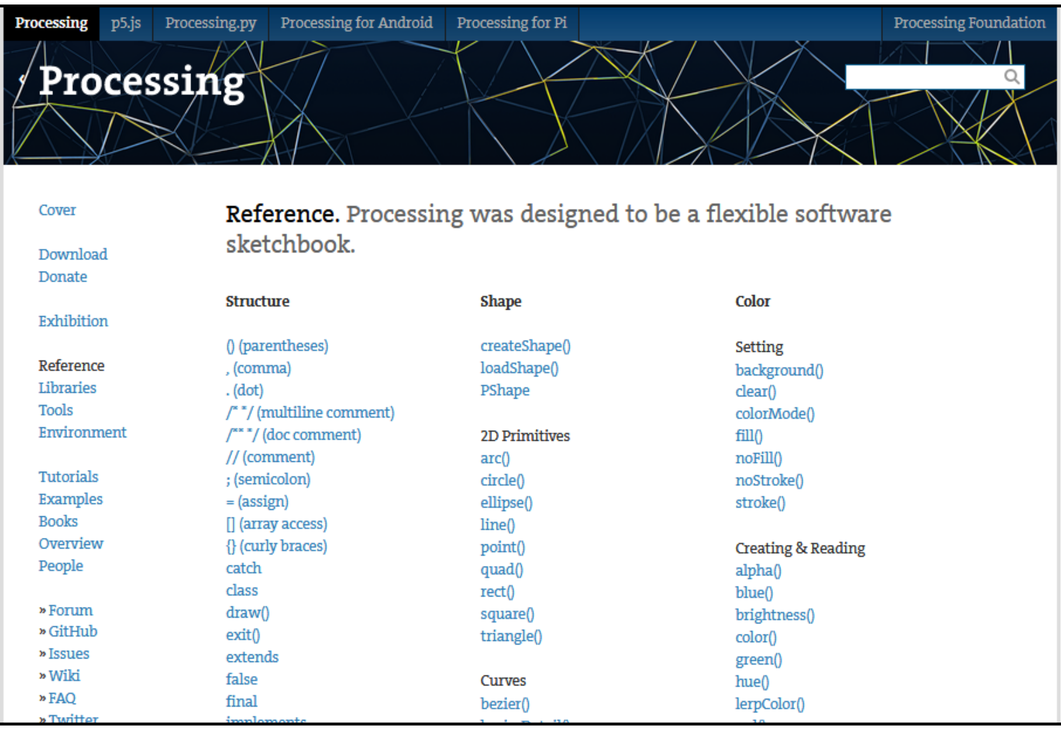
Processing 3.0 IDE



Primitive 2D Shapes

- point
- line
- triangle
- rect (rectangle)
- quad (quadrilateral, four-sided polygon)
- ellipse
- arc (section of an ellipse)
- curve (Catmull-Rom spline)
- bezier (Bezier curve)

31



Processing p5.js Processing.py Processing for Android Processing for Pi Processing Foundation

Processing

Cover

Download
Donate

Exhibition

Reference
Libraries
Tools
Environment

Tutorials
Examples
Books
Overview
People

» Forum
» GitHub
» Issues
» Wiki
» FAQ

Reference. Processing was designed to be a flexible software sketchbook.

Structure

() (parentheses)
, (comma)
. (dot)
/* */ (multiline comment)
/** */ (doc comment)
// (comment)
; (semicolon)
= (assign)
[] (array access)
{} (curly braces)
catch
class
draw()
exit()
extends
false
final

Shape

createShape()
loadShape()
PShape

2D Primitives
arc()
circle()
ellipse()
line()
point()
quad()
rect()
square()
triangle()

Curves
bezier()

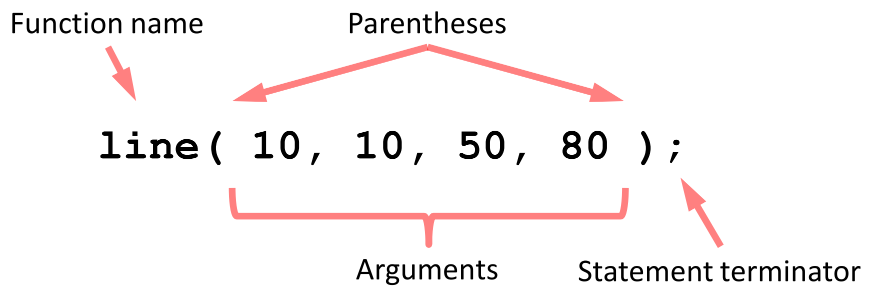
Color

Setting
background()
clear()
colorMode()
fill()
noFill()
noStroke()
stroke()

Creating & Reading
alpha()
blue()
brightness()
color()
green()
hue()
lerpColor()

32

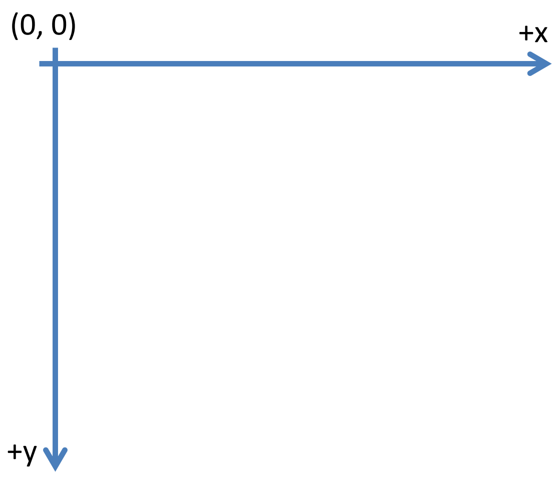
Anatomy of a Function Call



```
line( 10, 10, 50, 80 );
```

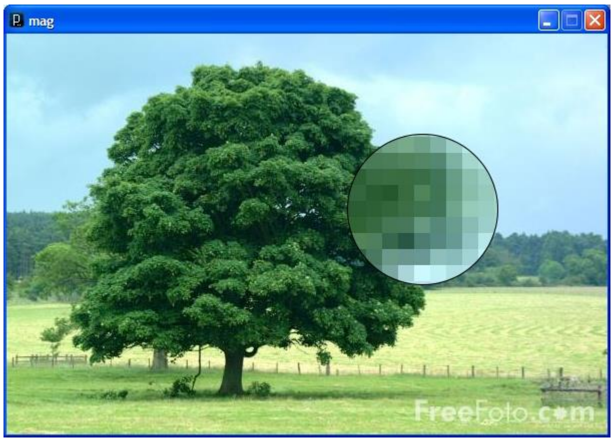
33

Coordinate System



34

Pixels



35

Processing Canvas

`size(` *`width, height`* **`);`**

Set the size of the canvas.

`background(` *`[0..255]`* **`);`**

Set the background grayscale color.

36

Drawing Primitives

```
point( x, y );

line( x1, y1, x2, y2 );

triangle( x1, y1, x2, y2, x3, y3 );

quad( x1, y1, x2, y2, x3, y3, x4, y4 );

rect( x, y width, height );

ellipse( x, y, width, height );
```

37

Colors

Composed of four elements:

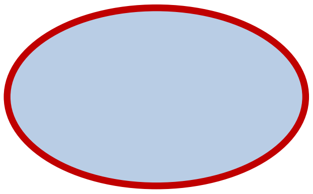
1. Red
2. Green
3. Blue
4. Alpha (Transparency)

38

Why 0 .. 255?

Shape Formatting

1. Fill color
2. Line thickness
3. Line color



These are properties of your paintbrush, not of the object you are painting.

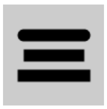
Fill Color

```
fill(gray);
fill(gray, alpha);
fill(red, green, blue);
fill(red, green, blue, alpha);

noFill();
```

Stroke (Line) Color

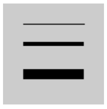
```
stroke(gray);
stroke(gray, alpha);
stroke(red, green, blue);
stroke(red, green, blue, alpha);

noStroke();
```

strokeCap()



```
smooth();
strokeWeight(12.0);
strokeCap(ROUND);
line(20, 30, 80, 30);
strokeCap(SQUARE);
line(20, 50, 80, 50);
strokeCap(PROJECT);
line(20, 70, 80, 70);
```

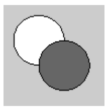
strokeWeight()



```
smooth();
strokeWeight(1);   // Default
line(20, 20, 80, 20);
strokeWeight(4);   // Thicker
line(20, 40, 80, 40);
strokeWeight(10);  // Beastly
line(20, 70, 80, 70);
```

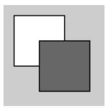
http://processing.org/reference/strokeCap_.html
http://processing.org/reference/strokeWeight_.html

ellipseMode



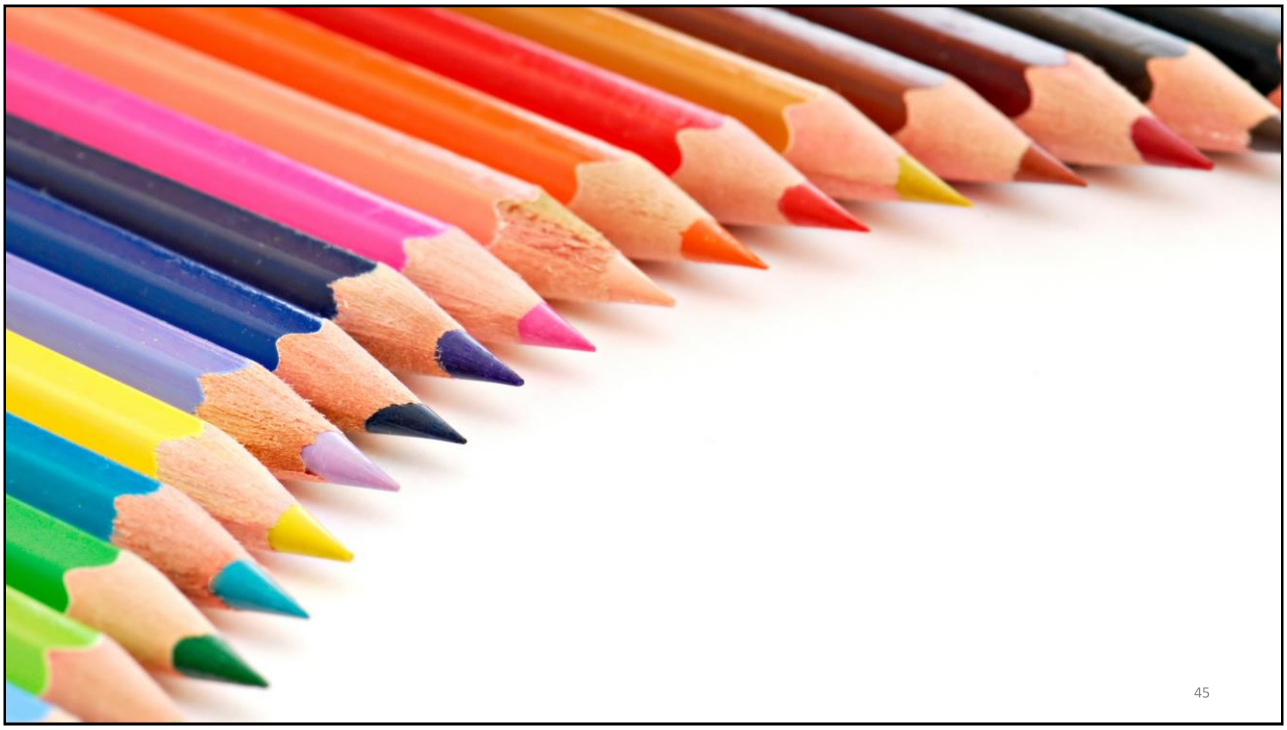
```
ellipseMode(CENTER);
ellipse(35, 35, 50, 50);
ellipseMode(CORNER);
fill(102);
ellipse(35, 35, 50, 50);
```

rectMode



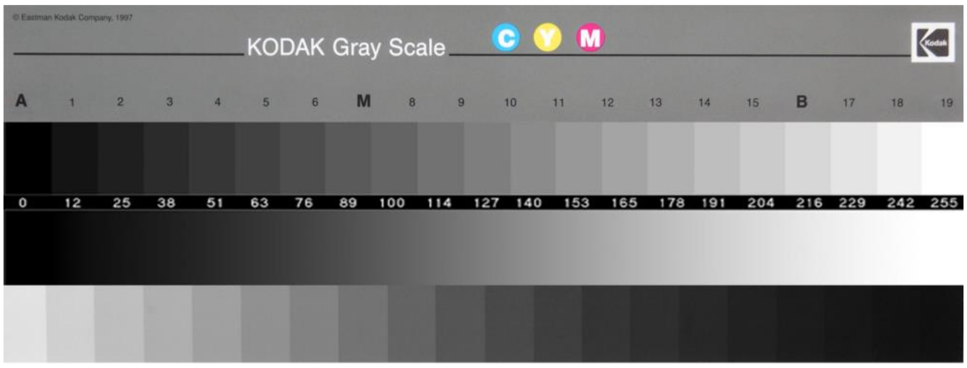
```
rectMode(CENTER);
rect(35, 35, 50, 50);
rectMode(CORNER);
fill(102);
rect(35, 35, 50, 50);
```

http://processing.org/reference/ellipseMode_.html
http://processing.org/reference/rectMode_.html



45

256 Shades of Gray



46