PROGRAMMING

INTERACTIVITY

PROCESSING

"EASY" PROGRAMMING TOOL FOR ARTISTS

STARTED IN 2001 BY FORMER MIT MEDIA LAB STUDENTS

SIMPLIFIED JAVA

PROCESSING.JS

"PORT" OF PROCESSING TO THE WEB BROWSER

ALMOST 100% COMPATIBLE

CAN BE RUN WITHOUT ANY ADDITIONAL SOFTWARE

PROCESSING RUNS SKETCHES

```
void setup(){
}
```

Runs once

```
void draw(){
}
```

Runs every frame

HTML PAGE

```
<!DOCTYPE html>
<head>
  <title>01_basics</title>
<body id="01_basics" onload="">
  <script type="text/javascript" src="./processing.js">
  </script>
  <canvas data-processing-
sources="01_basics.pde"></canvas>
</body>
```

DOWNLOAD PROCESSING.JS

http://bit.ly/1sPbOo1

```
// this is a comment
// setup is called just once
void setup(){
    // set our canvas size
    size(640,480);
    // set a yellow background color
    background(255,255,0);
    // set a black fill color
    fill(0,0,0);
}

// draw is called every time we paint the screen
void draw(){

   // draw a circle in the middle of the screen 50 pixel wide and tall
    // ellipse(x,y,w,h)
    ellipse(320,240,50,50);
}
```

Shapes

rect(x, y, w, h)
ellipse(x, y, w, h)
triangle(x1, y1, x2, y2, x3, y3)
line(x1, y1, x2, y2)
point(x, y)
arc(x, y, w, h, start, stop)
bezier(x1, y1, cx1, cy1, cx2, cy2, x2, y2)
quad(x1, y1, x2, y2, x3, y3, x4, y4)
image(image, x, y, width*, height*)

Colors

`background(r, g, b)`
Set the background color

`fill(r, g, b)`
Set the fill color for shapes

`noFill()`
Turn off fill for shapes

`stroke(r, g, b)`
Set the outline color for shapes

`strokeWeight(thickness)`
Change the thickness of lines and outlines

`noStroke()`
Turn off outlines for shapes

`color(r, g, b)`
Store a color in a variable

`blendColor(c1, c2, MODE)`
Blend two colors together

`lerpColor(c1, c2, amount)`
Find color between 2 colors

Text

`text(text, x, y)`
Draw some text
`textFont(font, size*)`
Changes the font of text

`textSize(size)`
Change the size of text

Environment

`width` / `height`
The size of the canvas

`draw = function() { };`
Called repeatedly during program execution.

`frameRate(fps)`
Change the frame rate of draw..

Keyboard

`key`
Number representing which key is pressed

`keyCode`
Represents when a special key is pressed

`keyIsPressed`
True if a key is being pressed, false otherwise

`keyPressed = function() { };`
Called when a key is pressed

`keyReleased = function() { };`
Called when a key is released

`keyTyped = function() { };`
Called when a key is typed

Mouse

mouseX, mouseY
Current coordinates of the mouse

pmouseX, pmouseY
Past coordinates of the mouse

mouseButton
Which button is pressed

mouseIsPressed
Whether mouse is being pressed

mouseClicked = function() { };
Called when mouse is clicked

mousePressed = function() { };
Called when mouse is pressed

mouseReleased = function() { };
Called when mouse is released

mouseMoved = function() { };
Called when mouse is moved

mouseDragged = function() { };
Called when mouse is released

mouseOver = function() { };
Called when mouse moves over canvas

mouseOut = function() { };
Called when mouse moves out of canvas

Transforms

`rotate(angle)`

Rotate shapes by an angle

`scale(amount)`

Scale shapes in both dimensions

`translate(x, y)`

Translate shapes by an offset

Math

`random(low, high)`
Generate a random number

`dist(x1, y1, x2, y2)`
Calculates the distance between two points

`constrain(value, min, max)`
Constrain value between min and max

`min(num1, num2)`
Return the minimum of two numbers
`max(num1, num2)`
Return the maximum of two numbers

`abs(num)`
Take the absolute value of a number
`log(num)`
Take the logarithm of a number
`pow(num, exponent)`
Raise a number to an exponent
`sq(num)`
Square a number
`sqrt(num)`
Take the square root of a number
`round(num)`
Return nearest whole number
`ceil(num)`
Return nearest whole number of greater/equal value
`floor(num)`
Return nearest whole number of lesser/equal value

ASSIGNMENTS

Make the game go slower

Add 3 lives to be used before game over

Add a score every time the paddle hits the ball

Add colored bricks to the top, that disappear when hit by the ball

Use the processing IDE and add sound effects

Extra credits: make another game

REFERENCES

- PROCESSINGJS.ORG
- PROCESSING.ORG
- OPENPROCESSING.ORG
- HTTPS://WWW.KHANACADEMY.ORG/COMPUTING/CS/PROGRAMMING