Week 04
Playing with Image Part 3

by Mike Wong. School of Creative Media

Quick Review of Part 2

More Methods of `PImage`

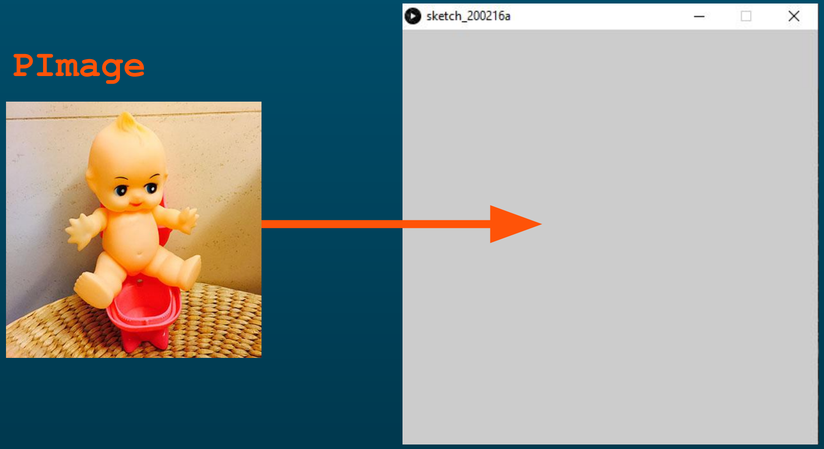
Methods	Description
`.copy()`	copies the entire image
`.mask()`	masks the image using another image as alpha channel
`.filter()`	filters the image using the specified filter
`.blend()`	blends the image with another one using various blending modes (PhotoShop alike)
`.save()`	Saves the image to a `.tiff`, `.tga`, `.png` or `.jpg` file

Review of Week 03 In-class Assignment

Playing with Image Part 3

Data transfer between Display and `PImage`

Display and PImage

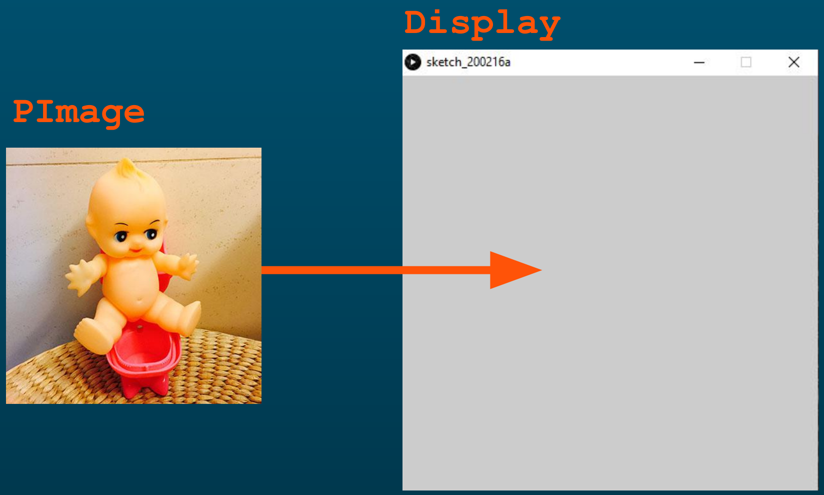


PImage

Display

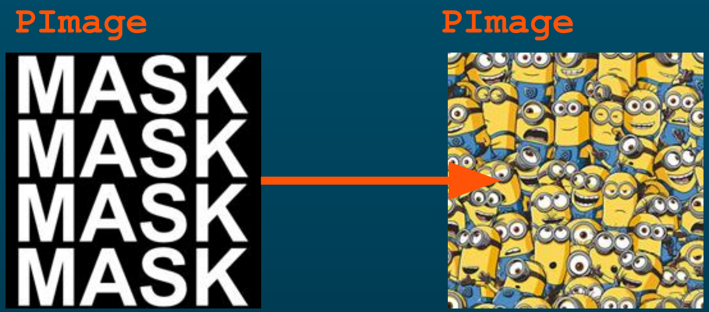
Part 1

Display and PImage



PImage

Display



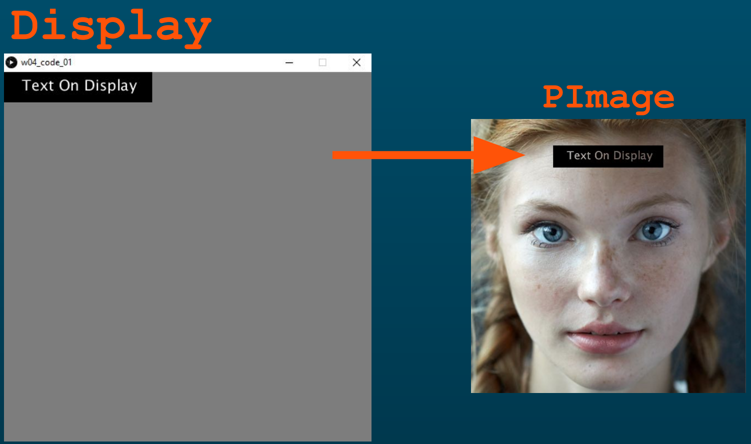
PImage

PImage

Part 1

Part 2

Display, PGraphics and PImage



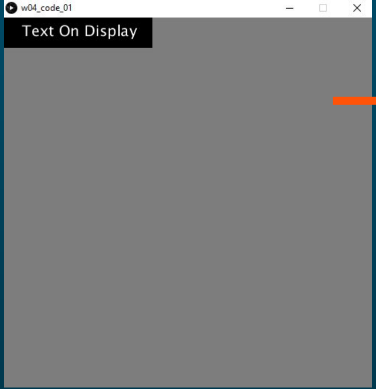
Display

PImage

Part 3

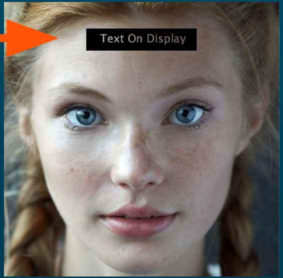
Display, PGraphics and PImage

Display


w04_code_01

Text On Display

PImage

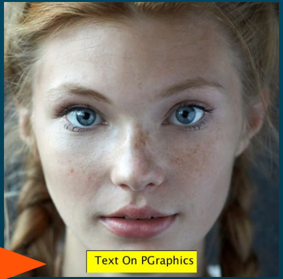


PGraphics

An off-screen (invisible) Display on which you may use most Processing Draw and Image functions.

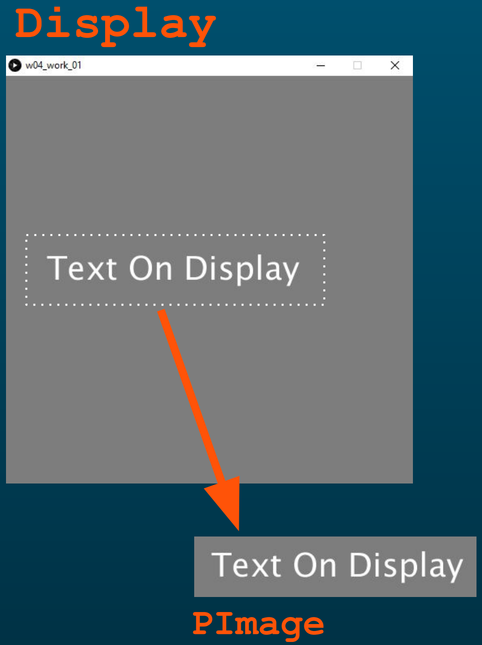
Text On PGraphics

PImage



Part 3

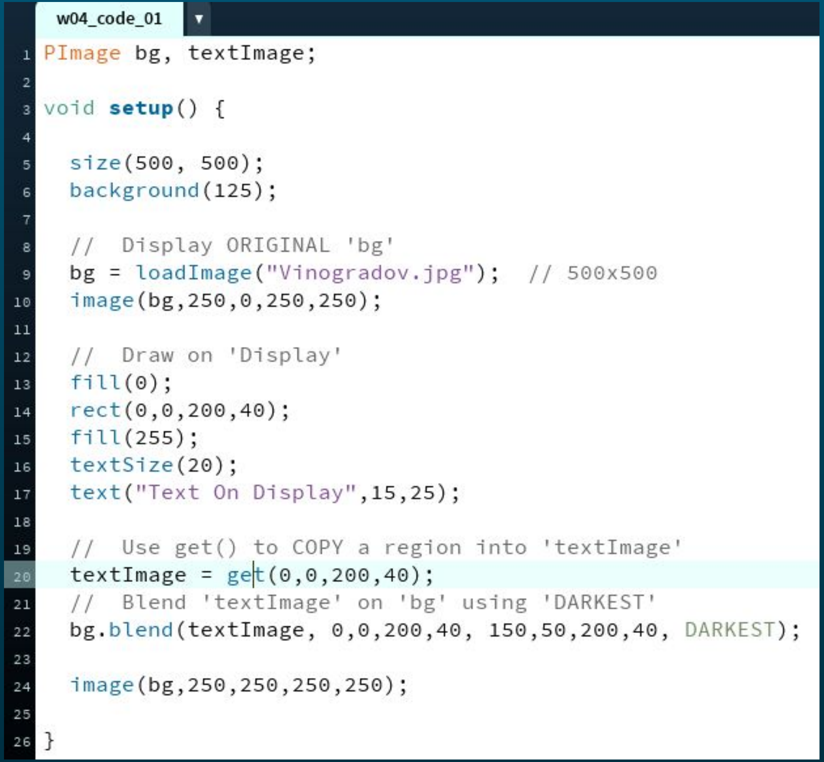
Display based get() function



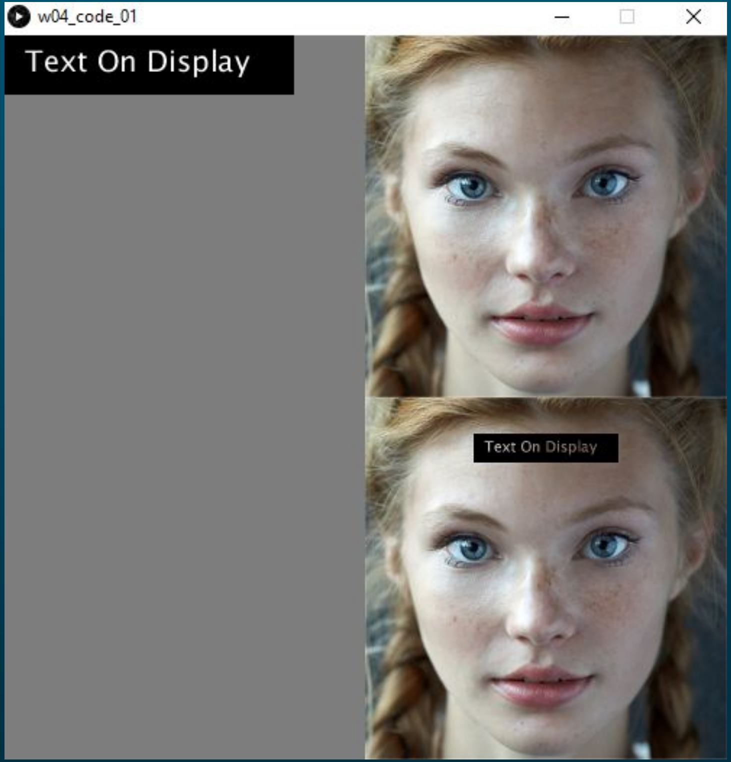
copies the whole or a region from the Display to as an PImage.

```
// returns the whole Display as an PImage
get();

// returns a region as an PImage
get(x,y,w,h);

// returns the color of pixel at (x,y)
get(x,y);
```

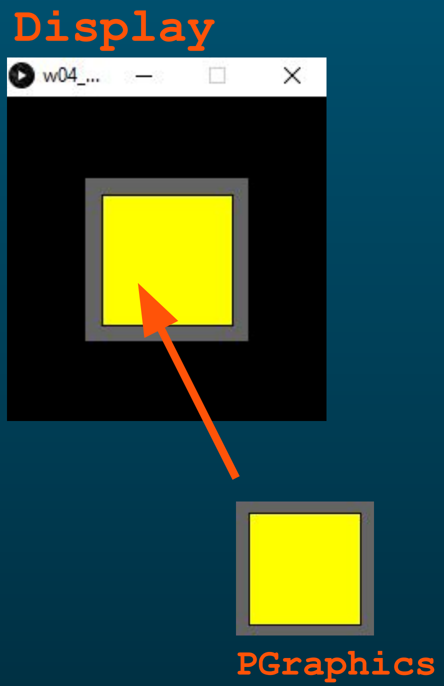
Example 01



```
PImage bg, textImage;

void setup() {

  size(500, 500);
  background(125);

  //  Display ORIGINAL 'bg'
  bg = loadImage("Vinogradov.jpg");  // 500x500
  image(bg,250,0,250,250);

  //  Draw on 'Display'
  fill(0);
  rect(0,0,200,40);
  fill(255);
  textSize(20);
  text("Text On Display",15,25);

  //  Use get() to COPY a region into 'textImage'
  textImage = get(0,0,200,40);
  //  Blend 'textImage' on 'bg' using 'DARKEST'
  bg.blend(textImage, 0,0,200,40, 150,50,200,40, DARKEST);

  image(bg,250,250,250,250);

}
```



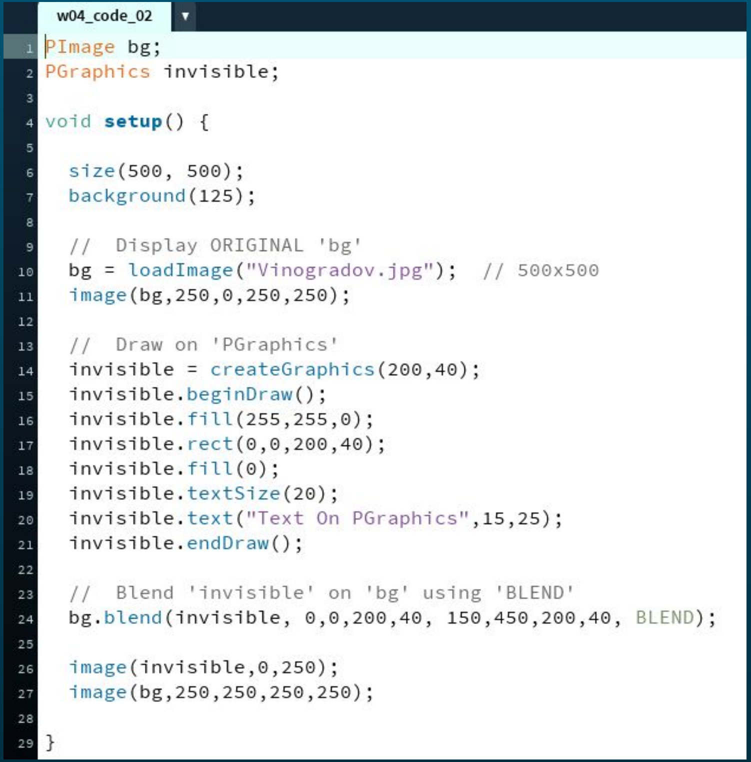
PGraphics An Off-screen Display



An Invisible display-like area on which we may use most Processing Draw and Image functions to create our contents. Its contents may then be referenced or displayed as a regular PImage.

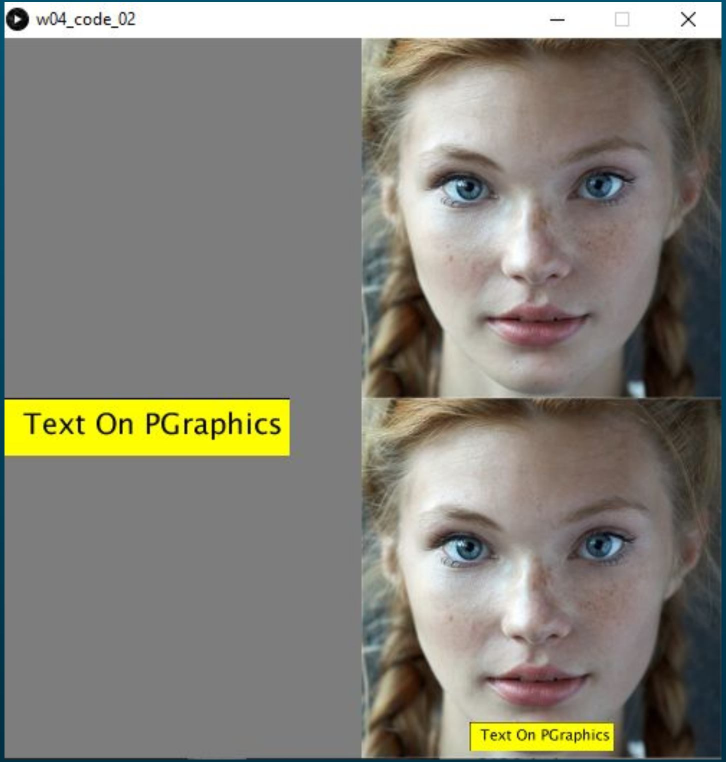
```
PGraphics pg;
void setup() {
  size(200, 200);
  background(0);
  pg = createGraphics(100, 100); // use createGraphics() to create
  pg.beginDraw();                // use .beginDraw() to start drawing
  pg.background(100);
  pg.fill(255,255,0);
  pg.rect(10,10,80,80);
  pg.endDraw();            // use .endDraw() to end drawing
  image(pg, 50, 50);       // can be displayed or manipulated as an PImage
}
```

Example 02

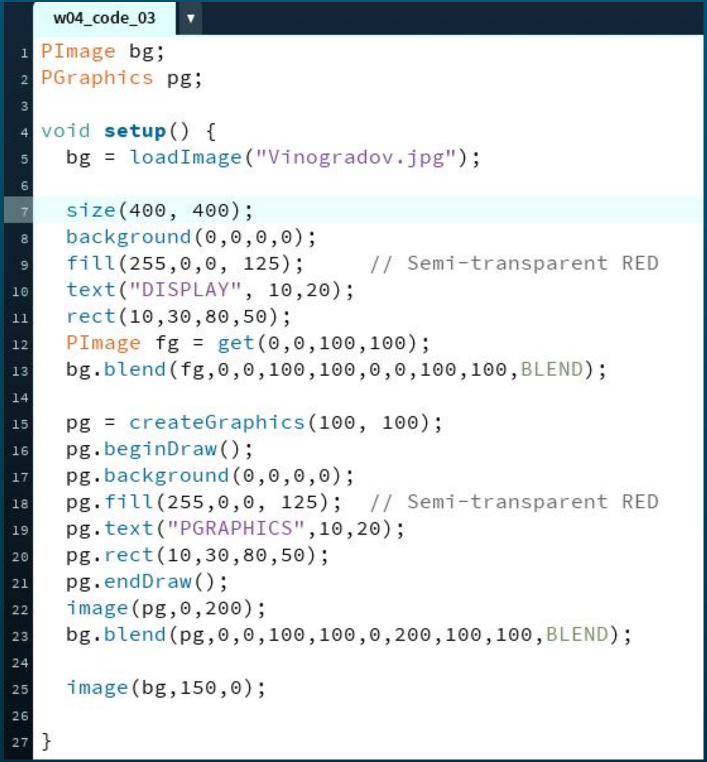


w04_code_02

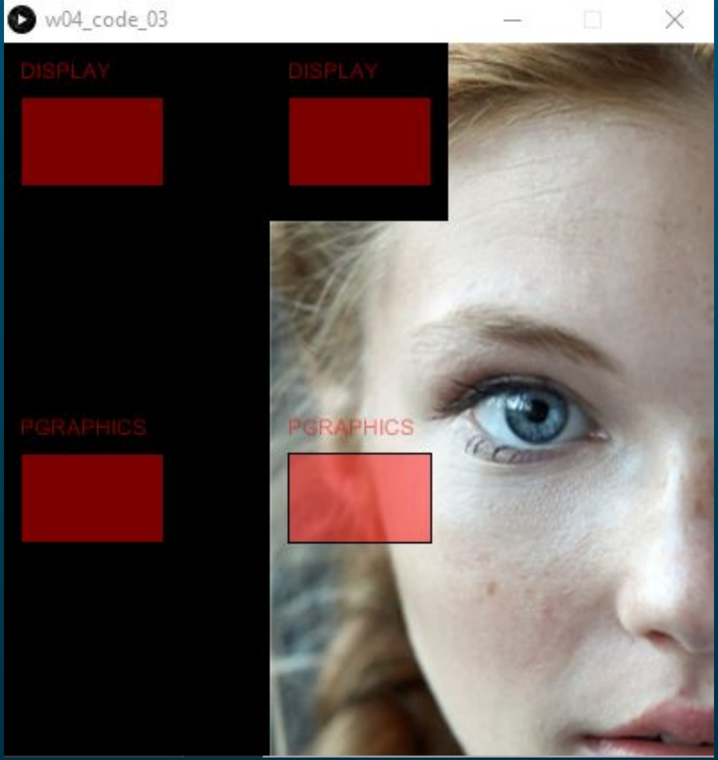
```
PImage bg;
PGraphics invisible;

void setup() {

  size(500, 500);
  background(125);

  //  Display ORIGINAL 'bg'
  bg = loadImage("Vinogradov.jpg");  // 500x500
  image(bg,250,0,250,250);

  //  Draw on 'PGraphics'
  invisible = createGraphics(200,40);
  invisible.beginDraw();
  invisible.fill(255,255,0);
  invisible.rect(0,0,200,40);
  invisible.fill(0);
  invisible.textSize(20);
  invisible.text("Text On PGraphics",15,25);
  invisible.endDraw();

  //  Blend 'invisible' on 'bg' using 'BLEND'
  bg.blend(invisible, 0,0,200,40, 150,450,200,40, BLEND);

  image(invisible,0,250);
  image(bg,250,250,250,250);

}
```



Example 03



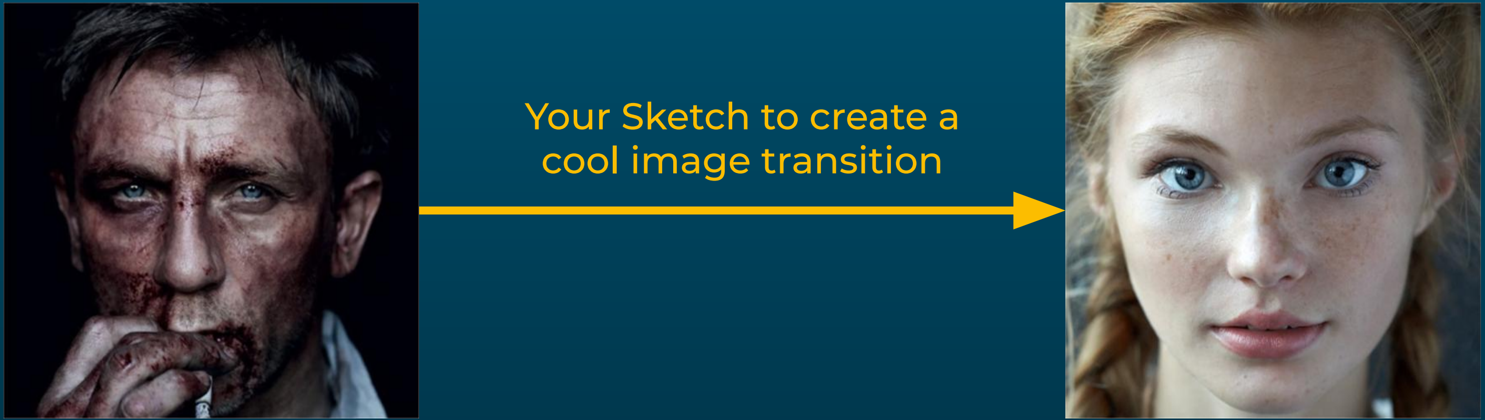
```
PImage bg;
PGraphics pg;

void setup() {
  bg = loadImage("Vinogradov.jpg");

  size(400, 400);
  background(0,0,0,0);
  fill(255,0,0, 125);      // Semi-transparent RED
  text("DISPLAY", 10,20);
  rect(10,30,80,50);
  PImage fg = get(0,0,100,100);
  bg.blend(fg,0,0,100,100,0,0,100,100,BLEND);

  pg = createGraphics(100, 100);
  pg.beginDraw();
  pg.background(0,0,0,0);
  pg.fill(255,0,0, 125);  // Semi-transparent RED
  pg.text("PGRAPHICS",10,20);
  pg.rect(10,30,80,50);
  pg.endDraw();
  image(pg,0,200);
  bg.blend(pg,0,0,100,100,0,200,100,100,BLEND);

  image(bg,150,0);

}
```



Other Display based functions

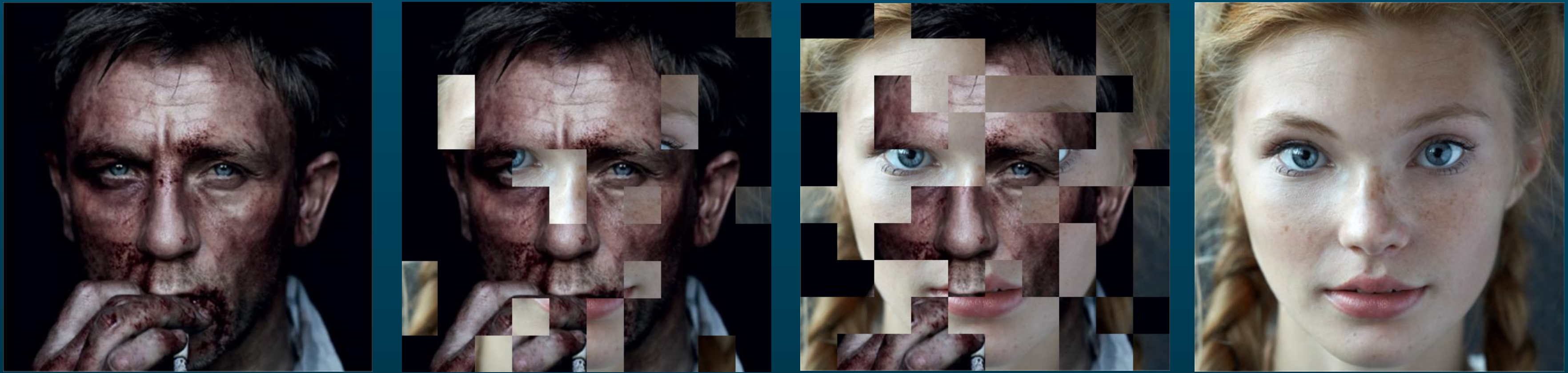
functions	Description
copy()	returns the entire display, intra-display region copy OR copy a region from a PImage
set()	set a pixel on display with the specified color or a region with the specified PImage
blend()	blends the display with a PImage using various blending modes (PhotoShop alike)
save()	Saves the display to a .tiff, .tga, .png or .jpg file

Assignment 01

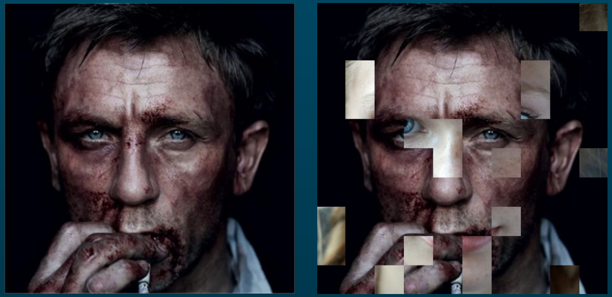


Your Sketch to create a cool image transition

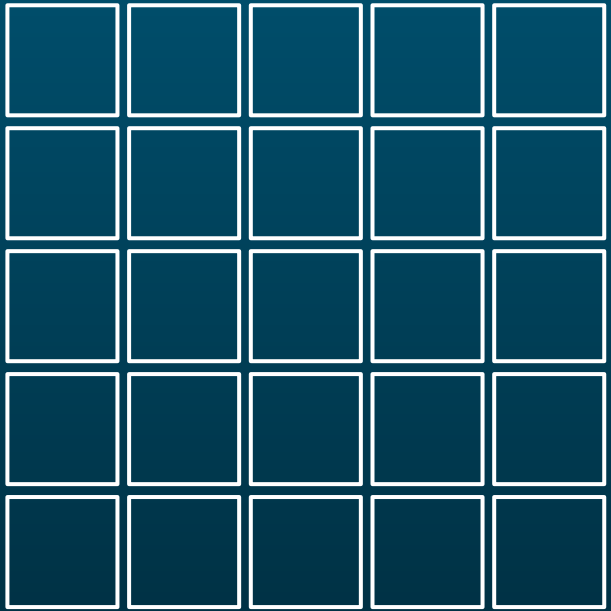
Creative Example 01



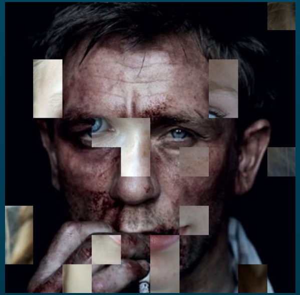
Creative Example 01



`numDiv = 5`



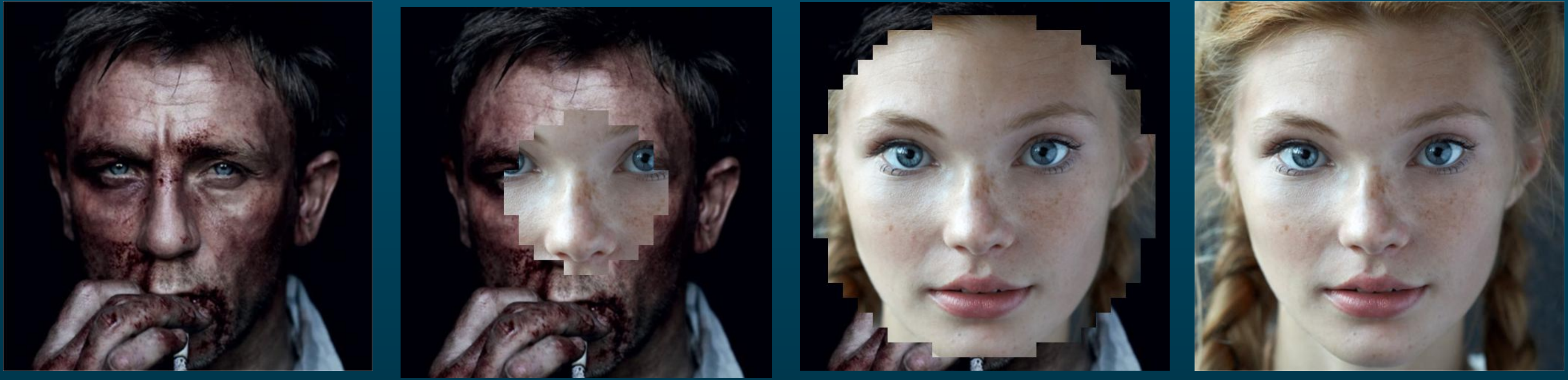
Creative Example 01



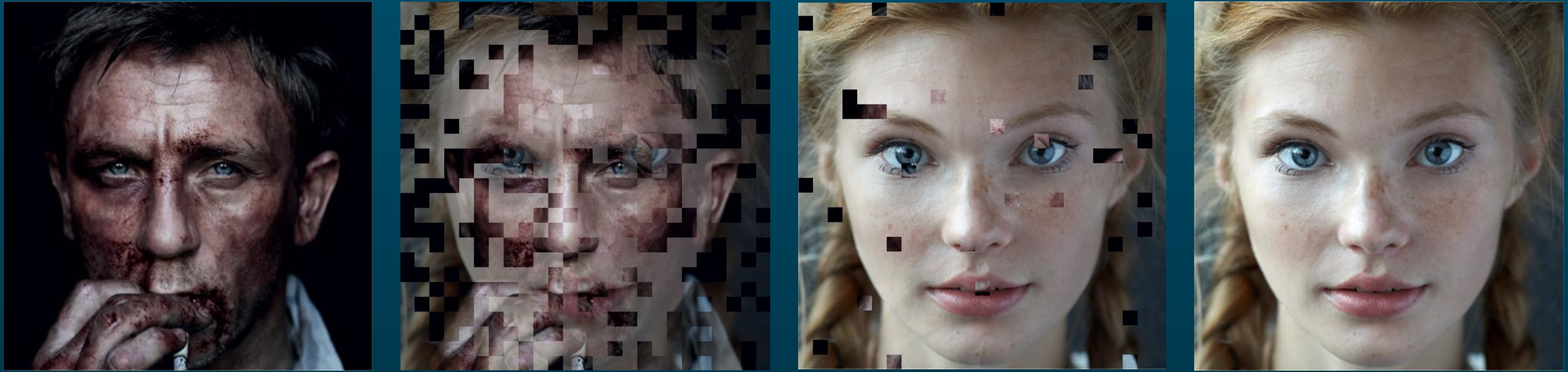
`Give each block a 'displayOrder'`
`e.g. from 0 to 7.`

0	3	5	4	1
7	2	1	7	6
1	4	0	3	2
5	0	2	6	3
2	3	5	1	4

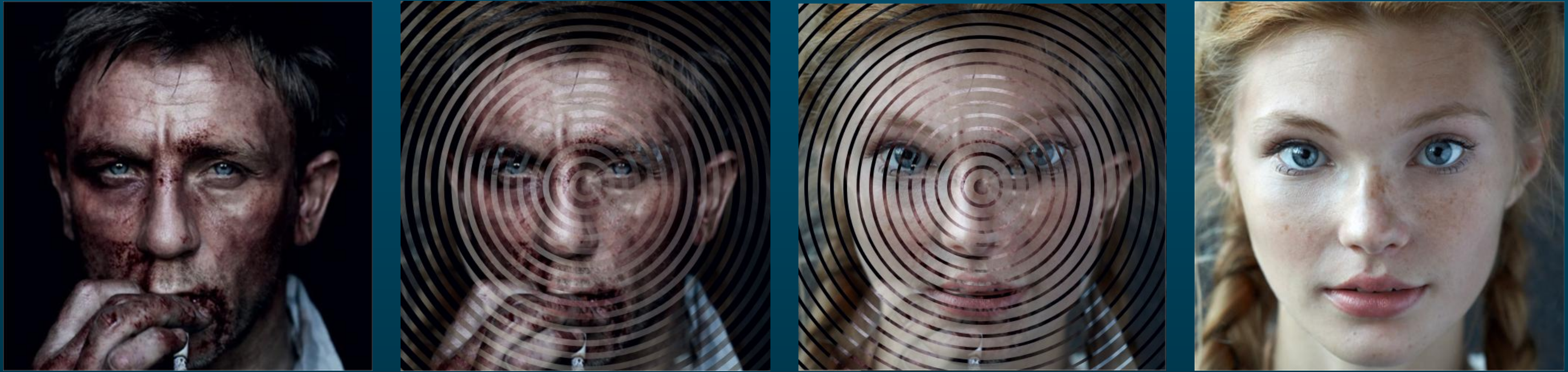
Creative Example 02



Creative Example 03 (PGraphics)



Creative Example 04 (PGraphics)



Creative Example 05 (PGraphics)