



sketch_160428a

Introduction à Processing

<http://www.processing.org>

Avril 2017

Historique

sketch_160428a

Design by Numbers

<http://dbn.media.mit.edu>

Date : 1999-2001

Lieu : MIT Media Lab

John Maeda



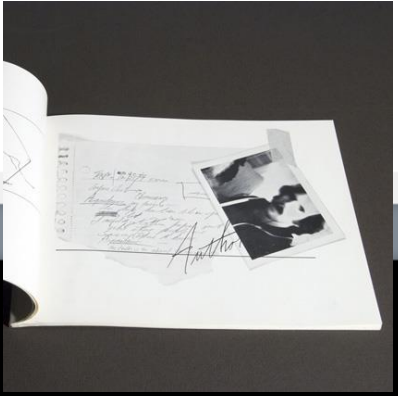
Visible Language Workshop

<http://museum.mit.edu/150/115>

Date : 1975

Lieu : MIT

Muriel Cooper



Processing



<http://www.processing.org>

Date : Printemps 2001

Lieu : MIT Media Lab

Ben Fry / Casey Reas



Processing 3



p5.js

Wiring

<http://wiring.org.co>

Date : 2003

Lieu : IDII

Hernando Barragán



Arduino



<http://www.arduino.cc>

Date : 2005

Lieu : IDII

Massimo Banzi





- Cover
- Download
- Donate
- Exhibition
- Reference
- Libraries
- Tools
- Environment

- Tutorials
- Examples
- Books
- Handbook

- Overview
- People

- Shop

- » Forum
- » GitHub
- » Issues
- » Wiki
- » FAQ
- » Twitter
- » Facebook
- » Medium

Download Processing. Processing is available for Linux, Mac OS X, and Windows. Select your choice to download the software below.



3.3.2 (25 April 2017)

Windows 64-bit
Windows 32-bit

Linux 64-bit
Linux 32-bit

Linux ARMv6hf
Linux ARM64

Mac OS X

- » [Github](#)
- » [Report Bugs](#)
- » [Wiki](#)
- » [Supported Platforms](#)

Read about the [changes in 3.0](#). The [list of revisions](#) covers the differences between releases in detail.

Stable Releases

- 3.3.2 (25 April 2017) [Win 32](#) / [Win 64](#) / [Linux 32](#) / [Linux 64](#) / [Linux ARMv6hf](#) / [Linux ARM64](#) / [Mac OS X](#)
- 3.3 (12 February 2017) [Win 32](#) / [Win 64](#) / [Linux 32](#) / [Linux 64](#) / [Linux ARMv6hf](#) / [Mac OS X](#)
- 2.2.1 (19 May 2014) [Win 32](#) / [Win 64](#) / [Linux 32](#) / [Linux 64](#) / [Mac OS X](#)
- 1.5.1 (15 May 2011) [Win \(standard\)](#) / [Win \(no Java\)](#) / [Linux x86](#) / [Mac OS X](#)

Earlier releases have been removed because we can only support the current versions of the software. To update old code, read the [changes](#) page. Changes for each release can be found in [revisions.txt](#). If you have problems with the current release, please [file a bug](#) so that we can fix it. Older releases can also be [built from the source](#). [Read More](#) about the releases and their numbering. To use [Android Mode](#), Processing 3 or later is required.

Qu

Processing : «

«

v

f

Processing est

il existe d'autr

même syntaxe

ed applications
th instant

r Android) mais
spectant la



sketch_160428a ▾





sketch_160428a

<https://www.youtube.com/shiffmanvideos>

<http://natureofcode.com>

Vidéos en ligne

Grid of video thumbnails for coding challenges:

- Coding Challenge #50.2: Animated Circle Packing - Part 2** (1 922 vues · il y a 5 heures)
- Coding Challenge #50.1: Animated Circle Packing - Part 1** (14 607 vues · il y a 1 jour)
- Coding Challenge #49: Photo Mosaic with White House Social Media Images** (22 647 vues · il y a 4 jours)
- Coding Challenge #48: White House Social Media Data Visualization** (12 056 vues · il y a 4 jours)

Programming from A to Z

Grid of video thumbnails for 'Programming from A to Z' series:

- Session 1: p5.js, JavaScript and Strings - Programming from A to Z** (Daniel Shiffman)
- Session 2: Regular Expressions - Programming from A to Z** (Daniel Shiffman)
- Session 3: Data, APIs and Language Processing Libraries...** (Daniel Shiffman)
- Session 4: Twitter API and Bots with Node.js - Programming from A to Z** (Daniel Shiffman)

p5.js tutorials - JavaScript, HTML, and CSS

Grid of video thumbnails for p5.js tutorials:

- 1-6: Foundations of Programming in JavaScript - p5.js Tutorial** (Daniel Shiffman)
- 7: HTML / CSS / DOM - p5.js Tutorial** (Daniel Shiffman)
- 8: Working with data - p5.js Tutorial** (Daniel Shiffman)
- 9: Additional Topics - p5.js Tutorial** (Daniel Shiffman)

Coding Challenges

Watch me take on some viewer submitted Coding Challenges in p5.js and Processing!

Grid of video thumbnails for coding challenges:

- Coding Challenge #1: Starfield in Processing** (13:54)
- Coding Challenge #2: Menger Sponge Fractal** (14:01)
- Coding Challenge #3: The Snake** (27:27)
- Coding Challenge #4: Purple Rain** (12:14)

Qu'est ce que Processing ?



sketch_160428a

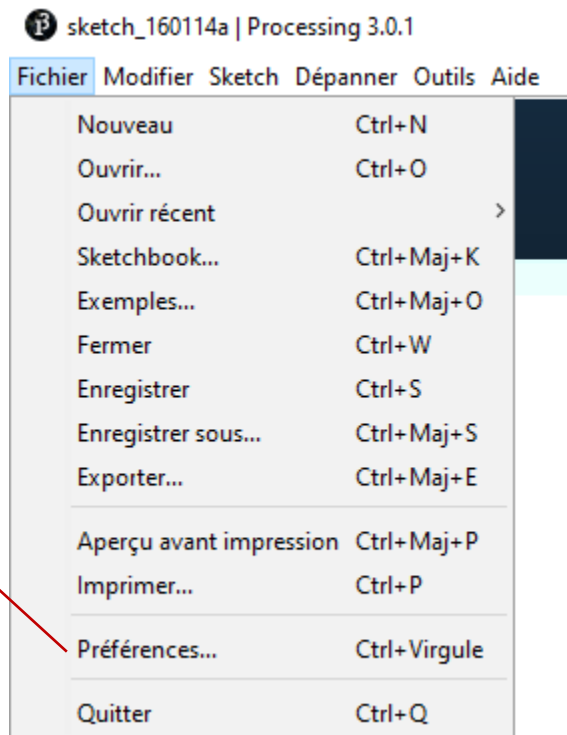
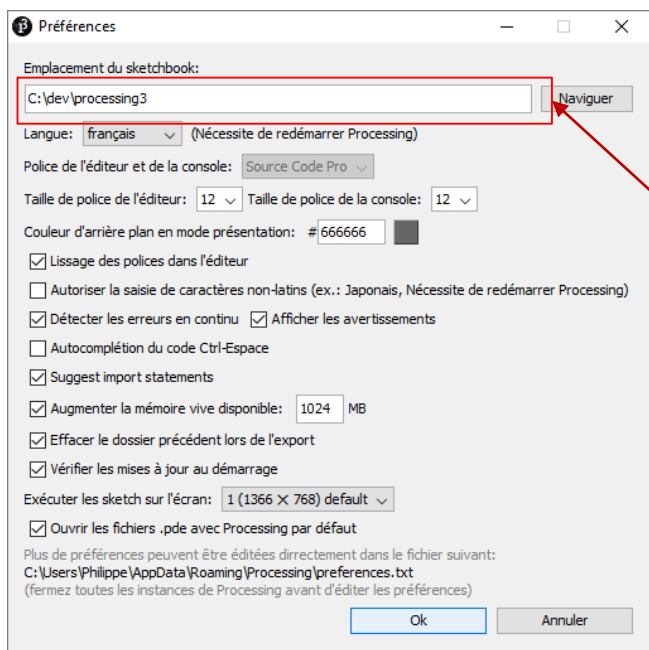
Arduino étant développé à la base à partir de Processing, il est simple de passer de l'un à l'autre pour prototyper des systèmes interactifs :

- **Processing** pour la partie interface et visualisation,
- **Arduino** pour la partie matérielle et gestion de capteurs.

La base du programme Processing est le « *sketch* » (programme, prototype)
L'extension est le « **.pde** »

Structure

- Les « sketches » (programmes) sont localisés dans le répertoire « préférences »



Structure

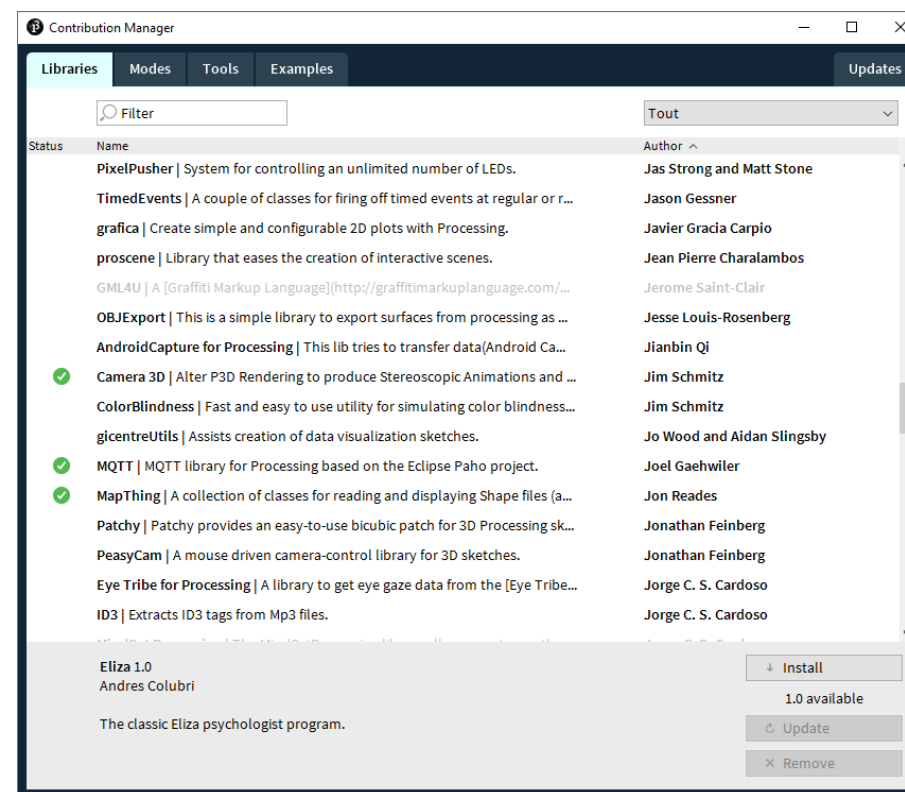
- un sketch est composé de :
 - Au moins un fichier « **.pde** » (cela peut être plus – un par classe objet).
Le fichier principal doit avoir le même nom que le répertoire du sketch
 - Plus d'autres répertoires
 - Répertoire « **data** » dans lequel se trouveront les ressources (images, sons, polices de caractères, ...)
 - Répertoire « **code** » dans lequel on peut trouver des librairies externes (fichier jar – archive java)

ELIPSE (C:) > dev > processing3 > boutons_img > Rechercher dans : boutons_i...

Nom	Modifié le	Type	Taille
code	05/12/2015 18:02	Dossier de fichiers	
data	06/12/2015 21:18	Dossier de fichiers	
boutons_img.pde	06/12/2015 21:47	Processing Source...	1 Ko
Button.pde	06/12/2015 21:47	Processing Source...	3 Ko

Structure

- Et de bibliothèques externes (Outils | Ajouter un Outil | onglet *Libraries*)



Deux fonctions basiques

- **setup** : exécuté une seule fois au démarrage – permet d’initialiser les variables du programme

```
void setup()  
{  
  size(200,200);  
  background(102);  
}
```

- **draw** : c’est la boucle de traitement et d’affichage exécutée « à l’infini » (*mainloop*)

Un premier exemple



sketch_160428a

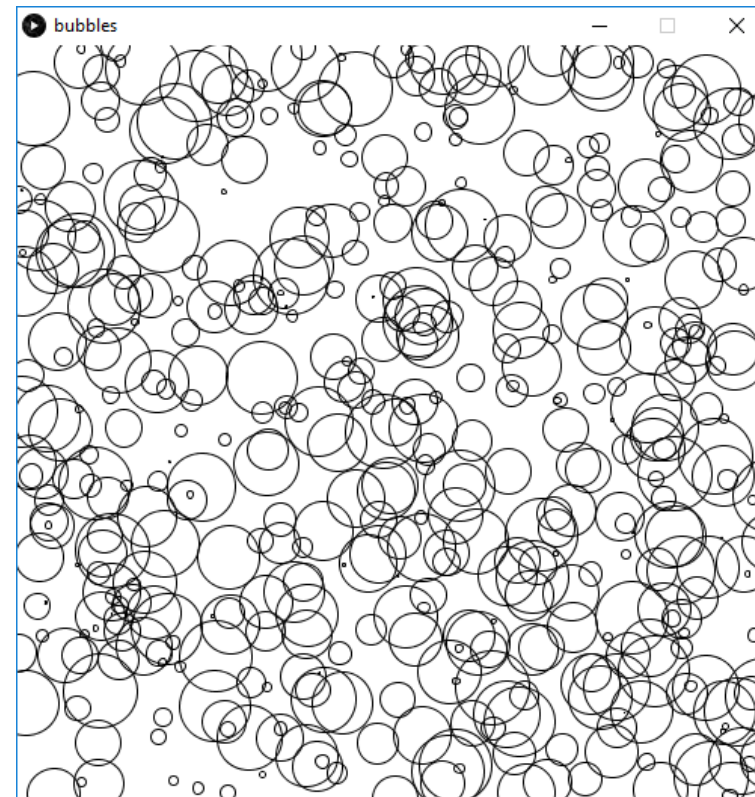
```
bubbles | Processing 3.3
Fichier Modifier Sketch Dépanner Outils Aide

bubbles
2 // variables du sketch (pseudos globales)
3 float x,y, diametre;
4
5 // setup n'est exécutée qu'une fois
6 void setup() {
7   size(500,500); // définit la taille de la fenêtre
8   background(255); // définit le fond de la fenêtre dessiné une fois (blanc)
9   smooth(); // anti aliasing
10  noFill(); // pas de remplissage des formes
11 }
12
13 // draw est exécutée à l'infini
14 void draw() {
15   x = random(width); // tiré au hasard entre 0 et la largeur de la fenêtre
16   y = random(height); // tiré au hasard entre 0 et la hauteur de la fenêtre
17   diametre = random(50);
18
19   // dessine un cercle - primitive pré-définie
20   ellipse(x,y, diametre, diametre);
21 }
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

L'enregistrement est terminé.

Console

Erreurs



Console

Erreurs

Exercices



sketch_160428a

- Notion de primitives graphiques
- Notion d'événements clavier/souris (*événements*)

- Utilisation de textes / images
- Utilisation de la webcam
- Utilisation de la réalité mixte
- Envoi/Réception d'informations de la liaison série

- *Aller plus loin* : écriture et usage de classes objets ...