



ON VA CODER !

Doc pour encadrants de l'atelier processing

Titouan Milon tmilon@student.42.fr

42 Staff pedago@staff.42.fr

Tuteurs tuteurs@student.42.fr

Résumé: Aucun jpeg n'a été maltraité pour la réalisation de ce sujet.

Table des matières

I	Introduction	2
II	Consignes générales	3
II.1	Consignes	3
II.2	Détails techniques	3



universcience

Chapitre I

Introduction

Bonjour à tous, tout d'abord merci de votre participation à cet atelier. Il ne sera pas en C contrairement à la majorité des projets 42, mais en Processing (du java simplifié orienté graphisme. Oui, je vous jure que ça peut être intéressant).

Le setup et l'atelier en lui-même sont assez simples, votre challenge principal sera d'encourager les élèves à réfléchir d'eux-mêmes sur les problèmes qui leur seront posés. Encouragez-les à échanger entre eux, si quelqu'un va trop vite, proposez-lui de travailler avec un élève ayant plus de difficultés. Privilégiez les métaphores pour expliquer la programmation, plus vos explications pourront se raccorder à des événements / objets de la vie de tous les jours, mieux ils comprendront.

Chapitre II

Consignes générales

II.1 Consignes

Pour setup l'atelier, il faut d'abord [installer Processing](#) si ce n'est pas déjà fait. Ensuite, si les fichiers de l'atelier ne sont pas disponibles sur la machine, vous pouvez les télécharger [ici](#). Il vous suffira ensuite d'ouvrir deux fenêtres, une contenant PAIMT.pde, et une contenant Demo_technique.pde, qui est un fichier avec tous les éléments dont les élèves auront besoin pour réaliser les premiers paliers. Le dossier solution est un guide rapide couvrant la majorité des exos du sujet pour vous dépanner si besoin (ne laissez pas les élèves y accéder !)

II.2 Détails techniques

Un programme Processing contient au strict minimum un fichier [nomChangeable.pde] se trouvant dans un dossier [nomChangeable] (ce nom pouvant être n'importe quoi, of course). D'autres fichiers .pde peuvent être ajoutés si besoin. Il faut au minimum une fonction setup() pour que le programme se lance. Pour ce qui est des images, elles doivent se trouver dans le dossier data du programme pour être load via loadImage(). Ce qui est demandé dans cet atelier ne devrait pas trop poser de problèmes de compréhension, et vous pouvez toujours vous aider de la doc en ligne et des exemples directement accessibles depuis l'IDE s'il y a quelque chose qui vous échappe.