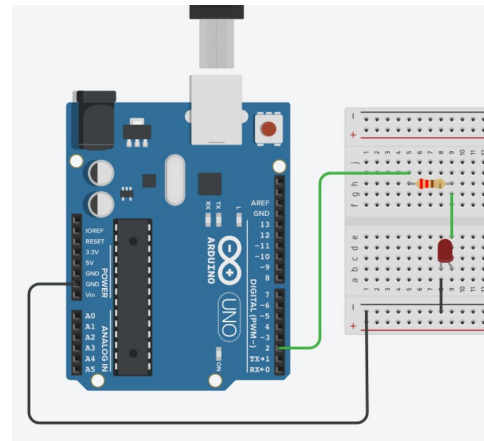
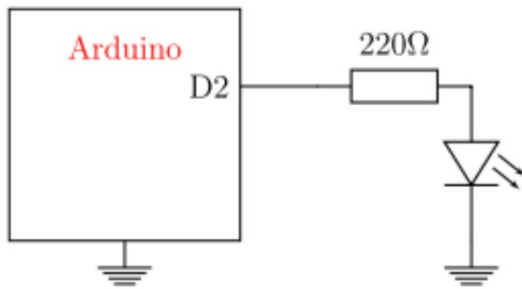


Suivre les instructions dans la vidéo pour les reproduire dans votre session Tinkercad : <https://scolawebtv.crdp-versailles.fr/?id=41130>

1 Première partie : allumer/éteindre

1.1 Une seule DEL

L'objectif est de simuler le montage ci-dessous à l'aide de la plateforme Tinkercad. La LED (libre choix de la couleur) s'allumera et s'éteindra toutes les secondes.



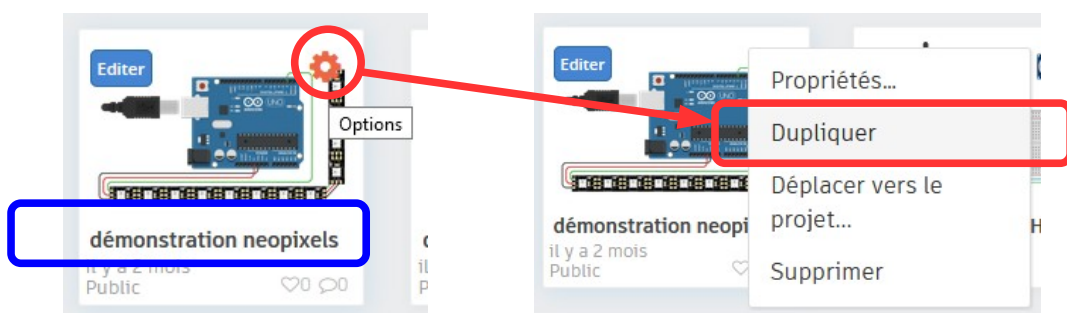
L'algorithme est le suivant :

```
Initialiser la broche D2 en sortie
Début
  Broche D2 ← NL1 { Allumée la led }
  Attendre 1s
  Broche D2 ← NL0 { Éteindre la led }
  Attendre 1s
Fin
```

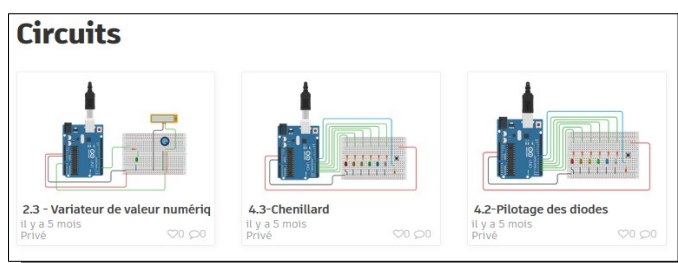
Travail à faire :

- faire le câblage sous Tinkercad,
- écrire les lignes de codes associées à l'algorithme,
- tester votre programme en lançant la simulation.

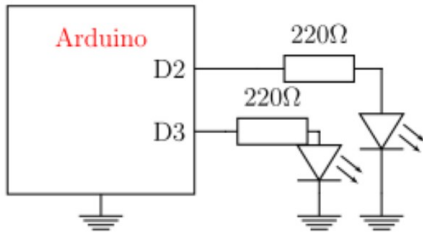
Ne pas écraser ses projets ! Afin de ne pas perdre de trace de son travail, et pouvoir revenir sur une version antérieure en cas d'erreur, il faut **dupliquer** son travail (pour enrichir la nouvelle version) et lui donner un **nom explicite** :



Exemple :



1.2 Deux DEL



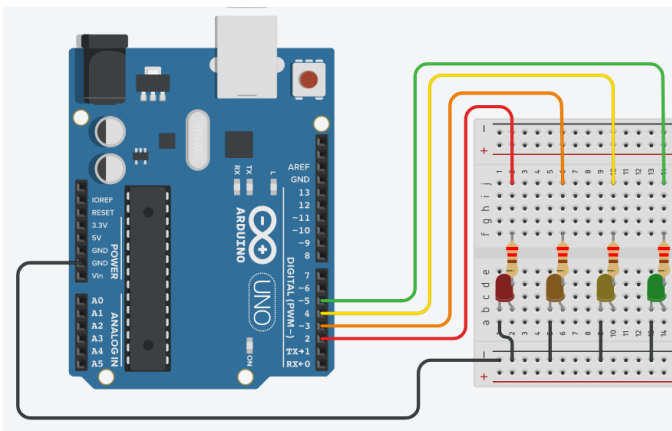
Les LED s'allumeront et s'éteindront alternativement toutes les secondes (quand l'une est allumée, l'autre doit être éteinte) :

- faire le câblage sous Tinkercad,
- écrire les lignes de codes associées à l'algorithme,
- tester votre programme en lançant la simulation.

1.3 Un chenillard de 4 DEL

Il vous faut simuler plusieurs fonctionnements du montage ci-dessous à l'aide de la plateforme Tinkercad.

1.3.1 Une lumière qui circule



Tout d'abord :

1. la led D2 s'allume,
2. temporisation de 1 s,
3. la led D2 s'éteint ET la led D3 s'allume,
4. temporisation de 1 s,
5. D3 s'éteint et D4 s'allume,
6. temporisation de 1 s,
7. on continue ainsi jusqu'à la led D5.

1.3.2 Une lumière qui circule... dans l'autre sens

Il vous faut simuler le fonctionnement du montage ci-dessous mais dans cet ordre-ci :

1. la led D5 s'allume,
2. temporisation de 1 s,
3. la led D5 s'éteint ET la led D4 s'allume,
4. temporisation de 1 s,
5. D4 s'éteint et D3 s'allume,
6. temporisation de 1 s,
7. on continue ainsi jusqu'à la led D2.

1.3.3 Un chenillard qui se remplit

Toujours avec le même montage, il vous faut simuler le fonctionnement suivant :

1. la led D2 s'allume,
2. temporisation de 1 s,
3. la led D2 reste allumée ET la led D3 s'allume,
4. temporisation de 1 s,
5. D2, D3 restent allumées et D4 s'allume,
6. temporisation de 1 s,
7. D2, D3, D4 restent allumées et D5 s'allume,
8. temporisation de 1 s,
9. on éteint toutes les leds,
10. temporisation de 1 s.

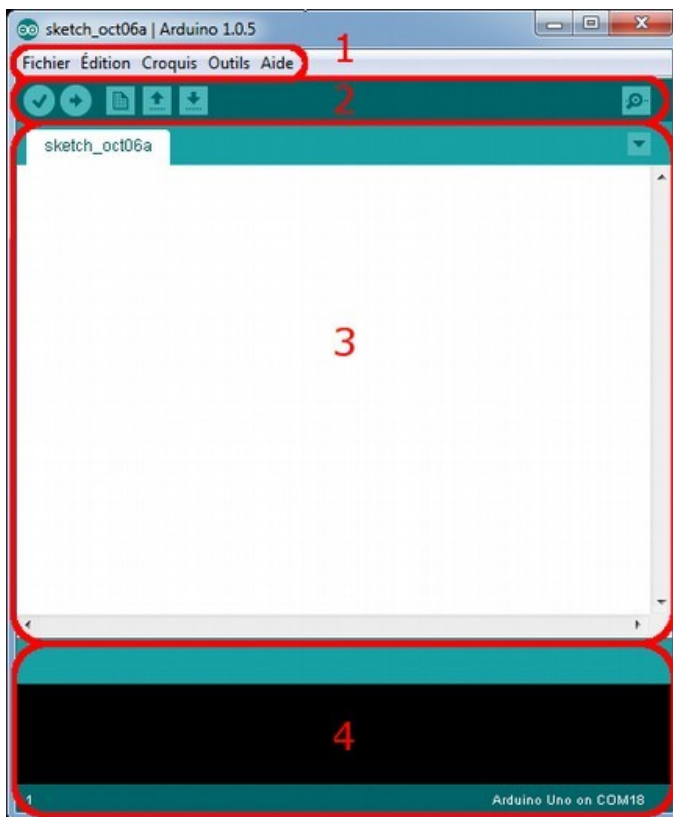
1.4 Utilisation du logiciel Arduino

Cadre numéro 1 : ce sont les options de configuration du logiciel.

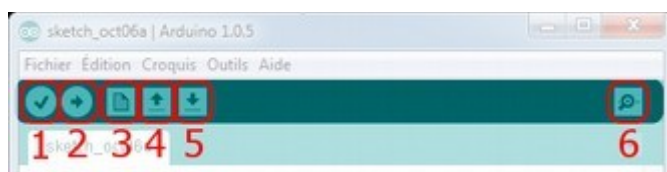
Cadre numéro 2 : il contient les boutons qui vont nous servir pour programmer les cartes Arduino.

Cadre numéro 3 : ce bloc va contenir le texte du programme à créer.

Cadre numéro 4 : celui-ci est important, car il va nous aider à corriger les fautes dans notre programme. C'est le débogueur.



Survoler les boutons de la zone 2 pour comprendre à quoi ils servent.

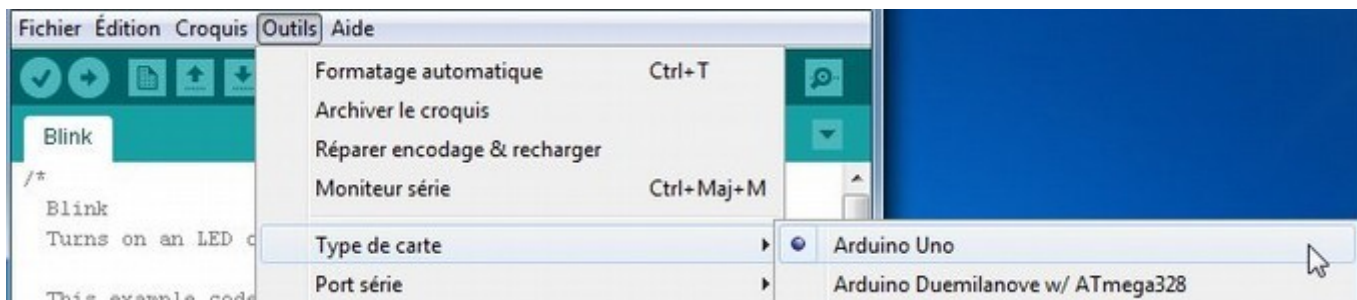


Taper le programme ci-contre, de façon rigoureuse :

Mais avant d'envoyer le programme dans la carte, il faut configurer le logiciel avec le nom de la carte et le port sur laquelle elle est branchée.

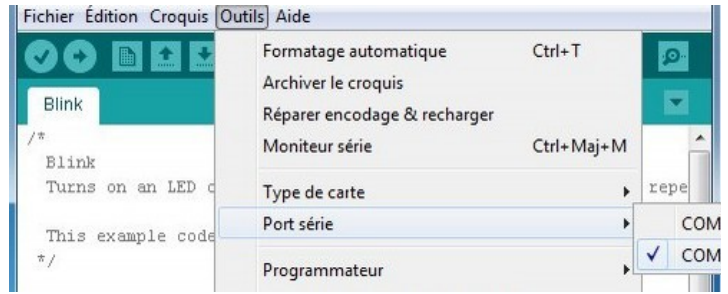
```
1 void setup() {
2   pinMode(13, OUTPUT);
3 }
4
5 void loop() {
6   digitalWrite(13, HIGH);
7   delay(1000);
8   digitalWrite(13, LOW);
9   delay(1000);
10 }
```

Choisir la carte que l'on va programmer : le nom de la carte est indiqué sur celle-ci, pour nous il s'agit de la carte "Arduino Nano". Dans le menu *Tools* ("outils" en français) puis dans *Board* ("Type de carte" en français), vérifier que c'est bien le nom "Arduino Uno" qui est coché. Si ce n'est pas le cas, la sélectionner :



Choisir le port de connexion de la carte. Aller dans le menu *Tools*, puis *Serial port* ("Port série" en français). Là, il faut choisir le port /dev/ttyACMx, x étant le numéro du port qui est affiché.

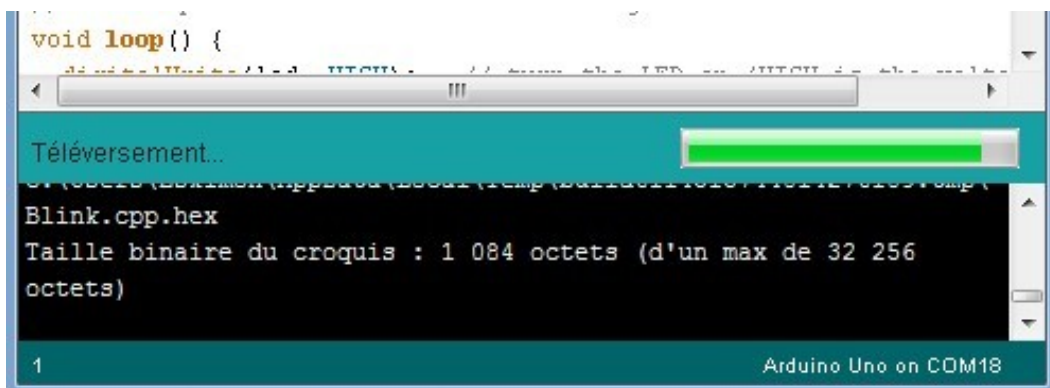
Dans l'exemple ci-contre, il s'agit de COM18 :



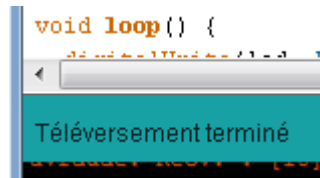
Maintenant, il va falloir envoyer le programme dans la carte. Pour cela il suffit de cliquer sur le bouton *Téléverser* (en jaune sur la photo) :



Ensuite apparaît le message "Compilation du croquis en cours..." pour informer l'utilisateur que le programme est vérifié puis compilé en langage machine avant d'être téléversé. Ensuite apparaît ce message-ci :



Une fois terminé, il affiche un dernier message :



Maintenant que le programme est dans la mémoire de la carte Arduino et qu'il est exécuté en boucle infinie, que se passe-t-il ? Que peut-on observer ?

Modifier les valeurs '13' & '1 000' et envoyer de nouveau le programme dans la carte pour essayer et comprendre.

1.5 [Du virtuel au réel](#)

Pour chacun des programmes créés dans Tinkercad, il faut :

1. câbler sur la platine seulement 4 DEL ;
2. copier le programme depuis Tinkercad et le coller dans l'éditeur Arduino ;
3. téléverser le programme dans la carte Arduino Nano ;
4. s'émerveiller de la magie de l'électronique !