

Lire le clavier & écrire sur l'écran du terminal – 10 avril 2022 / GL

Pour télécharger le programme 2_Terminal : **Clic droit** sur ce [lien](#) puis **Save link as**

Le placer sur votre bureau ou un dossier

Clic sur ce fichier 2_Terminal.ino qui va lancer l'IDE

L'IDE impose que le fichier .ino soit dans un dossier du même nom

Ce programme comprend 2 parties distinctes :

setup est exécuté une seule fois au lancement du programme

loop est exécuté indéfiniment (en boucle)

{ ... } indique début et fin de chaque partie (= procédure)

Tous les processeurs (du simple Arduino au complexe Pentium) exécutent séquentiellement des instructions
La boucle est le moyen le plus simple pour contrôler l'exécution des instructions

```
void setup() {  
  Serial.begin(115200);  
  Serial.println("Informatique = Rien de magique; tout est logique !");  
}  
  
void loop() {  
  byte car;  
  
  if (Serial.available() > 0) {  
    car = Serial.read();  
    Serial.println(car);  
  }  
}
```

Les explications ci-dessous font référence au **numéro de ligne** indiqué par l'IDE tout en bas :

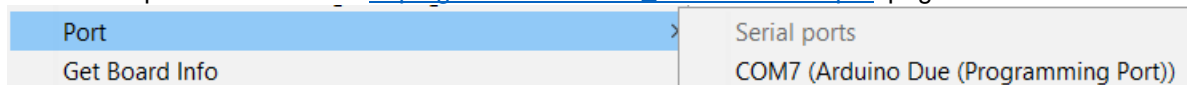
- 2 La communication entre PC et Arduino fonctionne à la vitesse (= débit binaire) de 115'200 bit/s
Ce qui correspond à environ 11520 caractères/seconde si chaque caractère utilise un format de 10 bit
- 3 Envoi du ("texte") : Informatique = Rien de magique; tout est logique !
- 7 Déclaration de la variable car locale à la procédure loop
Cette variable étant de type byte occupe 8 bit en mémoire
- 9 Test si caractère reçu
Résultat du type boolean = OUI / NON
Condition entre parenthèses ()
- 10 La variable car mémorise le caractère reçu (on dit qu'elle est affectée)
- 11 Le contenu de car est envoyé à l'écran

Le PC peut envoyer des caractères depuis l'IDE (Tools – Serial Monitor)

Je préfère utiliser putty comme terminal

Sauver son [exécutable](#) (clic-droit puis save link as)

Identifier le port USB utilisé → http://gelit.ch/Arduino/1_Familiarisation.pdf page 2



Configurer putty en conséquence

