**Java**

* **La généricité en Java**
* **Type<T>** variable
* **wildcard ?**, permet ici de recevoir n’importe quel type d'objet
* **? extends Voiture** : dire ici qu'on pourra utiliser les enfants de voitures et voiture
* **Fichier :**
* Les **FileInputStream :** sont destiner à la **lecture** dans le fichier
* Les **FileOutputStream :** sont destiner à l**’écriture** dans le fichier.
* Rôles :
* **DataInputStream**: offre la possibilité de lire directement des types primitifs (double,char,int) grâce à des méthodes : **readDouble(), readInt()…**
* **BufferedInputStream**: cette classe permet d'avoir un tampon à disposition dans la lecture du flux. En gros, les données vont tout d'abord remplir le tampon, et dès que celui-ci est plein, le programme accède aux données.
* **PushbackInputStream**: permet de remettre un octet déjà lu dans le flux entrant.
* **LineNumberInputStream**: cette classe offre la possibilité de récupérer le numéro de la ligne lue à un instant T.

**NB :** Ces classes prennent en paramètre une instance dérivant des classes **InputStream** (pour les classes héritant de **FilterInputStream**) ou de **OutputStream**(pour les classes héritant de **FilterOutputStream**)

* **La sérialisation**  c est le fait d’écrire un objet dans un fichier.
* La désérialisation d'un objet peut engendrer une **ClassNotFoundException**, pensez donc à la capturer avec **try…catch**

Lorsqu’ on utilise une Classe B dans une autre Classe A, et que la classe A doit être utiliser comme Objet pour charger un fichier(sérialisé), alors la classe B doit être :

* + - **Sérialisée ou**
    - **Dire à classe A que classe B ne doit pas être sérialiser** en ajoutant **transient lors de la declaration de l’objet B**
* **Declaration de la classe B(Notice)**
* **Declaration de la classe A(Game) qui est sérialisable :**
* Java 7 initie ce qu'on appelle vulgairement le **« try-with-resources »**. Ceci vous permet de déclarer les ressources utilisées directement dans le bloc Try(…), ces dernières seront automatiquement fermées à la fin du bloc d'instructions
* Pour des **ressources encapsuler** comme ceci :

On devra les décomposées :

**NB :** Pour rendre la fermeture automatique possible, les développeurs de la plateforme Java 7 ont créé une nouvelle interface :**java.lang.AutoCloseable.** Seuls les objets implémentant cette interface peuvent être utilisés de la sorte , **File** lui n implémente pas cette interface.