

# moteur

## intro:

Quand on utilise un moteur ce qu'on code est un controleur de moteur. On en utilise 4 différent en robotique: CANspark max, CANspark flex, Talon SRX, Talon FX.

CANspark max= moteur NEO

CANspark flex = moteur vortex

Talon FX = moteur Kraken

Talon SRX = autre moteur comme 775 pro ou SIM

## pour créer l'objet du moteur:

les 2 spark ce ressemble beaucoup.

exemple:

```
final SparkFlex lance = new SparkFlex(17 , MotorType.kBrushless);  
final SparkMax accumulateur = new SparkMax(18,  
MotorType.kBrushless);
```

final signifie qu'on ne peut modifier cette ligne a d'autre endroit dans le code

SparkFlex ou SparkMax le controleur a utiliser

lance est le nom du moteur c'est toi qui décide du nom

new SparkFlex signifie que tu veux en créer un nouveau

le 17 est l'ID c'est le nombre qui identifie ton moteur

MotorType.kBrushless ou MotorType.kBrush signifie que le moteur est soit avec ou sans broche.  
Les moteur a la robotique sont tous brusless

les 2 Talon ce ressemble beaucoup.

exemple:

```
final WPI_TalonSRX pince = new WPI_TalonSRX(2);
```

WPI\_TalonSRX ou WPI\_TalonFX le controleur a utiliser

pince est le nom du moteur

new WPI\_TalonSRX signifie que tu veux en créer un nouveau

le 2 est l'ID

ensuite pour dire quel utiliter donner au moteur on créé une fonction.

exemple:

```
public void lance(double vitesse){  
    Lanceur.set(vitesse);  
}
```

public signifie qu'il peut être utilisé partout dans le code

void signifie que tu retourne aucune valeur

le void pourrait être renplacer par:

double= renvoie un chiffre a virgule, int= renvoie un chiffre plein, boolean renvoie soit true ou false  
pour des objet a seulement 2 état par exemple, String= renvoie un message.

lance est le nom de la fonction elle sera utilisé dans les commandes (voir le chapitre sur les Commandes)

double vitesse est une variable qui pourra être modifier dans une commande

lanceur est le nom du moteur utiliser dans la fonction

set est ce que le moteur fait dans la fonction. Set mets la vitesse a la valeur mis dans les parenthèses en %

une fonction peut servir pour différentes choses mais on l'utilise principalement pour: définir la vitesse d'un moteur (comme l'exemple), prendre la valeur d'un encodeur (voir juste en bas, ou définir la position d'un encodeur (voir plus loin).

```
public double encodeur(){  
    return rotation.getEncoder().getPosition();  
}
```

public, double, encodeur et sont des mot qui on été défini plus haut

return signifie que ce qu'il y a après c'est ce que tu retourne comme valeur

rotatione.getEncoder().getPosition() est que tu prend la position de l'encodeur du moteur rotatione

pour définir la position d'un encodeur pour par exemple reset l'encodeur

```
public void reset(){  
    monte.getEncoder().setPosition(0);  
}
```

a la place de rotatione.getEncoder().getPosition() on met monte.getEncoder().setPosition() et le 0 est la valeur de l'encodeur qu'on définit

---

Revision #1

Created 11 April 2025 15:10:01 by Samuel Desharnais

Updated 11 April 2025 15:10:10 by Samuel Desharnais