

# MODULE 07

## SÉANCE WEB 03

### TP D'INFORMATIQUE

Durée 2h30

## CRÉER UNE TRAJECTOIRE

### UNITÉ CERTIFICATIVE

U6 - VALORISATION DE LA DONNÉE ET CYBERSÉCURITÉ

### COMPÉTENCE(S)

C08 - CODER

### OBJECTIF PÉDAGOGIQUE

Créer une API qui permet de récupérer les données de vols au format JSON et de les enregistrer dans une base de données

### CONNAISSANCES ISSUES DU RÉFÉRENTIEL

- Langages de développement, de description, de création d'API et les IDE associés Niveau 4
- Chaînes de développements (ordinateur, embarqué, cross compilation) Niveau 3

### CONNAISSANCES OPÉRATIONNALISÉES

- Codage d'une page HTML avec une feuille de style CSS Niveau 2

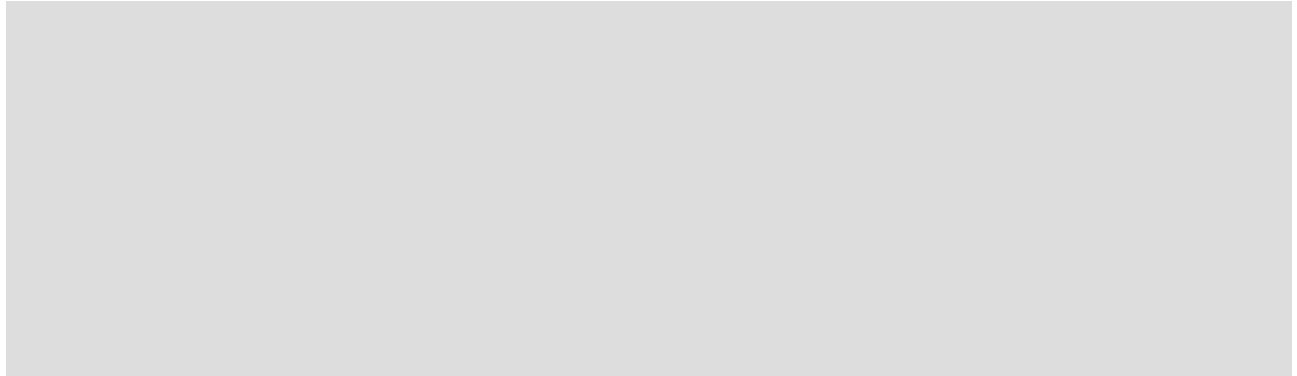
## TD

# Dessiner dans un canvas

### Tracer de lignes

Pour tracer des lignes, nous allons utiliser les méthodes « **moveTo**, **lineTo** et **stroke** »

En vous aidant du net, donner l'utilité de chaque méthode



Donner le code complet permettant de tracer une ligne vertical d'une valeur de 100 à partir de la position (x,y)

```
ctx.moveTo(x, y);  
ctx.lineTo(x, 100);
```

Donner le code complet permettant de tracer une ligne horizontal d'une valeur de 100 à partir de la position (x,y)

```
ctx.moveTo(x, y);  
ctx.lineTo(100,y);
```

### Tracer une croix

Maintenant nous souaitons tracer une croix dont les diagonales mesures 10 à partir de la position (x,y)

Donner le code permettant de tracer la croix précédente

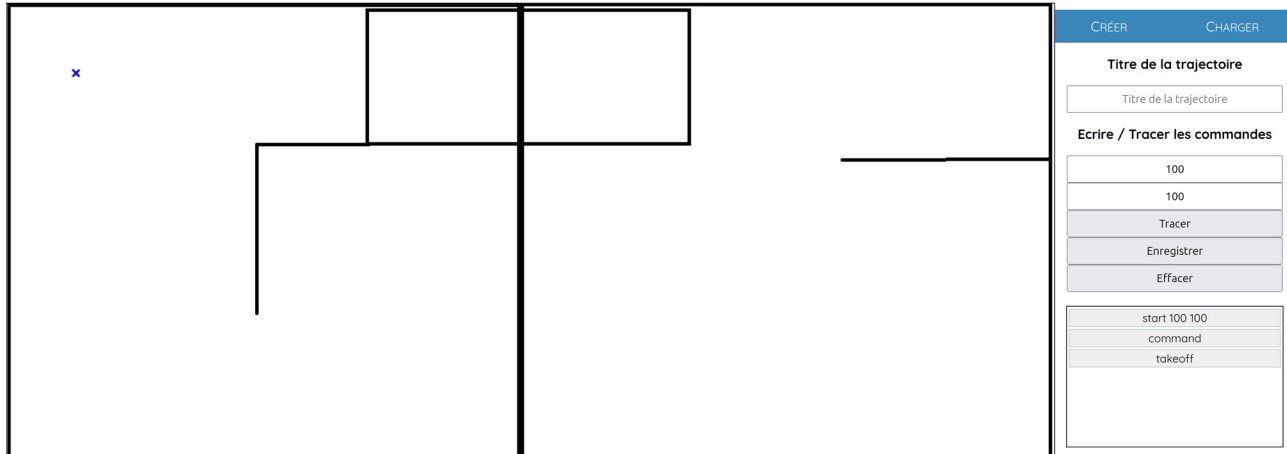
```
ctx.moveTo(x, y);  
ctx.moveTo(x-5, y-5);  
ctx.lineTo(x+5, y+5);  
ctx.moveTo(x+5, y-5);  
ctx.lineTo(x-5, y+5);
```

## Défi 1

### Point de départ

#### Objectif

Positionner et tracer le point de départ du drone.



#### Récupérer les valeurs x et y

Pour connaître le point de départ, nous allons utiliser la première valeur des champs x et y comme position de départ

Créer une variable globale « **isStartingPosition** » qui vaut false. Cette variable vaudra « true » après avoir positionné le point de départ

Créer une fonction « **dessinerTrajectoire(x,y)** »

Compléter la fonction pour tracer une croix à la position (x,y) d'un longueur de 10 si la valeur de « **isStartingPosition** » vaut false

#### Tracer la croix de départ

Rendre le bouton « Tracer » cliquable dans la fonction « **initialisationNavTrajectoire()** »

Le bouton précédent devra récupérer la valeur de x et de y puis appeler la fonction « **dessinerTrajectoire(x,y)** »

Vérifier l'affichage d'une croix sur le canvas.

#### Afficher les commandes

Compléter le fichier js avec une fonction permettant de lister dans la zone de texte (sous le bouton effacer, le texte comme dans la figure précédente

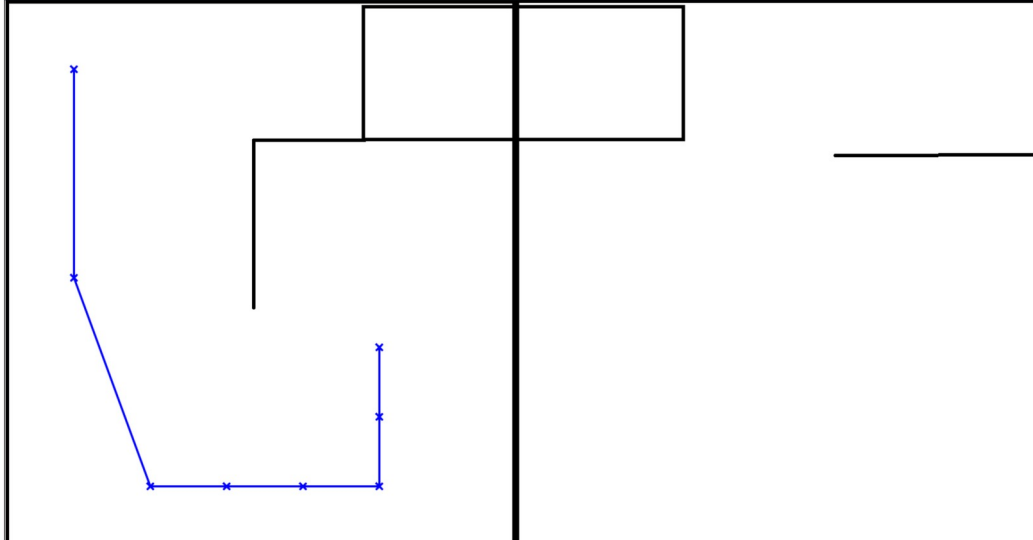
## Défi 2

### Tracer les lignes

#### Objectif :

Tracer des lignes et une croix en fonction des valeurs de x et y.

Attention à bien respecter les limites extérieures



CRÉER CHARGER

Titre de la trajectoire

Titre de la trajectoire

Ecrire / Tracer les commandes

0

-100

Tracer

Enregistrer

Effacer

start 100 100

command

takeoff

go 0 300 0 50

go 110 300 0 50

go 110 0 0 50

go 110 0 0 50

go 110 0 0 50

go 0 -100 0 50

go 0 -100 0 50

#### Mise en place

Sur le même principe que l'étape précédente, compléter le fichier js afin de tracer les lignes, les croix et la liste des commandes à chaque fois que l'utilisateur cliquera sur le bouton tracer.

N.B. : Pour les commandes, nous utiliserons la valeur de « go x y 0 50 »

N.B. : Pour tracer la ligne suivante il faut partir de la dernière position

## Points de départ

Le bouton cliquable doit récupérer la valeur des champs « x » et « y » puis appeler la fonction « dessinerTrajectoire(x,y) »

Une fois le chemin tracé, les valeurs de x et y sont gardées dans les variables globales « lastX et lastY »

Puis on écrit la commande « go x y 0 50 » dans la zone de texte sous