

# Three.js

## Et géographie

**Christophe Vestri**

Le mardi 1 février 2022

# Plan du cours

- 7 janvier : Intro, github, Capteur/Geoloc en HTML5
- 14 janvier: carto/geo, leaflet/mapBox, rest Api
- 25 janvier: 2D/3D: Canvas, WebGL et Three.js
- 1 février: Aframe/AR.js, exercice + projet
- ~~8 février~~ -> 22 février: Projets

# Plan Cours 4

- Rappel dernier cours
- Questions Three.js et réponses
- Exercice:
  - Three.js + Géolocalisation + Rest Api
  - Projet

# Html5

- **Acces capteur caméra:**
- **DeviceOrientation, DeviceMotion**
- **Caméra, Audio, Géolocalisation**
- **touchevents/mouse/...**
- **<https://developers.google.com/web/fundamentals/native-hardware/device-orientation/>**

# Leafletjs

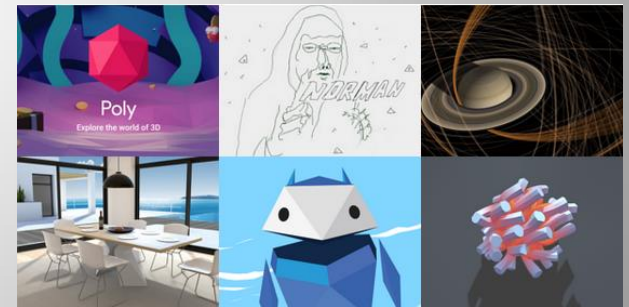
- [leafletjs](https://leafletjs.com/) est une librairie Opensource pour afficher des cartes interactives utiles à la navigation (comme google maps)
- Seulement 33Ko, Tous les browsers
  - Map controls
  - Layers
  - Interaction Features
  - Custom maps



# Three.js

The logo for Three.js, featuring the word "THREE" in white serif font and ".js" in a smaller white sans-serif font, both on a red rectangular background.

- Qu'est-ce que Three.js
  - Couche abstraite et haut niveau de WebGL
  - Librairie javascript pour créer des scènes 3D
  - Cross-plateforme et gratuit
  - Rendus en webGL, CSS3D et SVG
  - <https://threejs.org/>
  - <https://davidlyons.dev/threejs-intro/>



# Questions/Solution Three.js

- Texture et modèles
  - Il faut [serveur local](#)
- OrbitControl
  - Attention aux workflow modules -> sinon [JS files link](#)
- Modèles 3D
  - [GLTFLoader](#) ou autre
  - Ne pas oublier l'éclairage

# Outils de debug

- En local:
  - `python3 -m http.server`
  - <http://localhost:8000/> firefox ou chrome
- Smartphone android -> Chrome
- <https://developers.google.com/web/tools/chrome-devtools/javascript>
  - Simulation de smartphone (F12)
  - Connecté à un smartphone: <chrome://inspect/>
- Firefox possible ou autres??



# Three.js

- **Exercice (2h) : Geolocalisation sur terre**
  - Créez une scène + caméra + light + renderer
  - Créez une sphère de rayon 1
  - Texturez cet objet avec image de planète terre
  - Créer une fonction Lat/Lon to cartésien
  - Récupérer votre position et afficher un marqueur
  - Récupérer les positions de plusieurs pays et afficher des marqueurs (couleur différente)
  - <https://restcountries.com/v3.1/all>
  - mettre un modèle 3D à la place du marqueur à notre position, drapeau du pays

# Projet final

Evaluation:

Exos des cours (50%)

Projet (50%)

- **Projet final**
  - Capteurs mouvement/orientation
  - Géolocalisation et/ou objets geolocalisés
  - UI et scene 3D, interaction
  - Exemples:
    - Compas 2D/3D: carte 2D + geoloc et directions 3D
    - Objets 3D animés avec interaction smartphone
- **Présentation**
  - Qqs slides, 5/10min chacun avec démo sur écran/smartphone