

# Atelier Solidity

## Objectif

- i. Préparation de l'environnement de développement
- ii. Déploiement et interaction avec un smart contrat
- iii. Manipulation des comptes, des événements, et des fonctions du contrat

## 1<sup>ère</sup> partie : Préparation de l'environnement de développement

Installer :

- Node.js
- Truffle
- Ganache
- Visual Studio Code
- Extension Solidity (dans VS Code)

## 2<sup>ème</sup> partie : Création d'un nouveau projet

1. Créer un nouveau projet truffle
2. Créer un smart Contract :

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract Message {
    uint public number;
    event NumberUpdated(uint oldNumber, uint newNumber);

    function set(uint _num) public {
        emit NumberUpdated(number, _num);
        number = _num;
    }
    function get() public view returns (uint) {
        return number;
    }
}
```

3. Déployer le contrat

```
// migrations/2_deploy_message.js
const Message = artifacts.require("Message");
module.exports = function (deployer) {
    deployer.deploy(Message);
};
```

4. Lancer Ganache, puis ajouter un nouveau projet en sélection le fichier *truffle-config.js*.
5. Ajouter la configuration suivante au fichier *truffle-config.js*.

```
networks: {  
  
  ganache: {  
    host: "127.0.0.1",      // Localhost (default: none)  
    port: 7545,             // Standard Ethereum port (default: none)  
    network_id: "*",       // Any network (default: none)  
  },  
}
```

## 6. Déployer le smart contract

### 3<sup>ème</sup> Interagir avec le contrat

- a. Lancer la console Truffle
- b. Lister les comptes
- c. Déployer et interagir
- d. Observer les événements