

# JSLIGO

HELLO WORLD

Langage de smart contrat inspiré par Typescript qui compile vers Michelson

```
type storage = string;
type parameter = unit ;
type return_ = [list<operation>, storage];

const main = (action: parameter, store: storage) : return_ =>
  [list([]), "Hello Tezos!"];
```

# JSLIGO

## MOVEMENTS DE LINK

```
type direction = {kind : "North"} | {kind:"East"} | {kind:"South"} | {kind:"West"};
type parameter = {kind: "Face", direction: direction}
                | {kind: "Start"}
                | {kind:"Stop"};

type state = {kind : "Idle"} | {kind:"Moving"};
type storage =
{
  state: state,
  orientation: direction
};

type return_ = [list <operation>, storage];

const main = (action: parameter, store: storage) : return_ => {
  const noop = list([]);
  switch(action.kind){
    case "Face": return [noop, {state: {kind : "Idle"}, orientation: : action.direction}];
    case "Start": return [noop, {...store, state: {kind:"Moving"}}];
    case "Stop": return [noop, {...store, state: {kind : "Idle"}}];
  }
};
```