

MÉMOIRE.JS

```
//-----LE SYSTEME DE POINTAGE-----\\

if(!localStorage.getItem("Meilleurs-Scores")){

    localStorage.setItem("Meilleurs-Scores", JSON.stringify([{"Score": 0, "Moment": Date.now()}, {"Score": 0, "Moment": Date.now()}, {"Score": 0, "Moment": Date.now()}]));

}

//Mettre à jour le score par comparaison\\

function verifier_score(score){

    let scores = JSON.parse(localStorage.getItem("Meilleurs-Scores"));

    if(scores[0].Score < score){

        scores[2].Score = scores[1].Score;

        scores[2].Moment = scores[1].Moment;

        scores[1].Score = scores[0].Score;

        scores[1].Moment = scores[0].Moment;

        scores[0].Score = score;

        scores[0].Moment = Date.now();

    }else if(scores[1].Score < score){

        scores[2].Score = scores[1].Score;

        scores[2].Moment = scores[1].Moment;

        scores[1].Score = score;

        scores[1].Moment = Date.now();

    }else if(scores[2].Score < score){

        scores[2].Score = score;

        scores[2].Moment = Date.now();

    }

    localStorage.setItem("Meilleurs-Scores", JSON.stringify(scores));

}
```

```

//-----VARIABLES-----\\

//Les couleurs d'arriere plan
var CouleurAP = ["lightblue", "lightblue", "lightblue", "#45B114", "#ffb703", "#fb8500", "#F40D0D"];

//La fenetre d'entree pour la longueur du qudrillage
var entree = document.getElementById("entree");
var btnPlus = document.getElementById("btn-plus");
var btnMoins = document.getElementById("btn-moins");

//bouton pour demarer, continuer et recommencer le jeu
var btnDepart = document.getElementById("btn-depart");
var btnContinuer = document.getElementById("btn-continuer");
var btnRecommencer = document.getElementById("btn-recommencer");

//tableau qui contiendra les tuiles generees
var tuiles = [];

//le jeu doit etre active par le bouton depart
var jouer = false;

/**
 * On commence au niveau
 * Les tuiles sont gardés dans le tableau «boutonsDeJeu»
 * Le premier tuile de la sequence a memoriser est à la position 0
 * Le nombre actuel fait référence au sous-niveau joué
 * Le score est aussi définit de façon globale
 */

var niveau = 1;
var boutonsDeJeu = [];
var position = 0;
var actuel = 1;
var score;

//-----AU DÉMARRAGE-----\\

```

```

//--//DEMARRAGE\\--\\
window.onload = function(){
    var nombre = document.getElementById("nombre");

    entree.style.display = "block";

    btnPlus.addEventListener("click", () => nombre.innerText = (parseInt(nombre.innerText) + 1) <= 5?
    parseInt(nombre.innerText) + 1 : 5);

    btnMoins.addEventListener("click", () => nombre.innerText = (parseInt(nombre.innerText) - 1) >= 2?
    parseInt(nombre.innerText) - 1 : 2);
}

//-----MÉCANISME DES TUILES-----\\
//déclenche l'animation d'une tuile
function clickBtn(boutton, delai){
    bouton.style.animation = "activation 0.7s 1";
    timeoutID = setTimeout(()=>boutton.style.animation = "", delai);
}

//-----ALGORITHMES DE HASARD-----\\
//Algorithme de hasard, maximum exclusif & minimum inclusif
const nbrHasard = (min, max) => Math.floor(Math.random() * (max - min) + min);
//Crée la séquence
function tableauHasard(longueur, grandeur){
    let tableau = [];

    //on doit retrancher 1 de grandeur^2 car le tableau commence à 0
    for(let i = 0; i<longueur; i++){tableau.push(nbrHasard(0, grandeur*grandeur-1));}

    return tableau;
}

//Modifier le curseur lorsque les tuiles sont générées
function modifie_curseur(curseur){
    document.querySelectorAll(".tuile").forEach(element => element.setAttribute("style",
    "cursor:"+curseur));
}

```

```

}

//Répéter le div d'un tuile dans une rangée donnée
const repete_mot = (mot, compte) => compte > 1? mot + repete_mot(mot, compte-1) : mot;

//----GÉNÉRER LES TUILES-----\\
document.getElementById("btn-go").addEventListener("click", ()=>{
    entree.style.display = "none";
    //--//Générer le quadrillage\\--\\
    for(let i=0; i<nombre.innerText; i++){
        document.getElementById("jeu").innerHTML += `<div class="rangée"> ${repete_mot('<div
class="tuile"></div>', nombre.innerText)} </div>`;
        //--//Tableau contenant les boutons\\--\\
        tuiles = Array.from(document.getElementsByClassName("tuile"));
    }
    /**Écouter un clique,
    * vérifier si le jeu est en marche
    * si oui, vérifier si c'est la bonne tuile
    **/
    tuiles.forEach((tuile)=>{
        tuile.addEventListener("click", (e)=>{
            if(jouer == false){
                return false;
            }
            clickBtn(e.target, 750);
            //--shift() enlève & renvoie le premier element d'un tableau--\\
            if(tuiles.indexOf(e.target) != boutonsDeJeu.shift()){
                verifier_score(score);
                setTimeout(()=>{
                    //Écran d'échec

```

```

        document.getElementById("echec").style.display = "block";

        document.getElementById("echec-entete").innerHTML = `<h1>DOMMAGE, VOUS AVEZ
PERDU AU NIVEAU ${niveau}</h1>`;

        document.getElementById("echec-corps").innerHTML = `<h3>Votre meilleur score est
${JSON.parse(localStorage.getItem("Meilleurs-Scores"))[0].Score}`;

        niveau = 1;

        modifie_curseur("default");

    }, 1100);

    btnDepart.disabled = false;

    jouer = false;

} else {

    score += nbrHasard(85, 126) + 10 * position;

    position++;

    if (boutonsDeJeu.length == 0) {

        setTimeout(() => {

            score += Math.pow(10, niveau - 1);

            verifier_score(score);

            if (actuel % niveau != 0) {

                //Écran fin de ronde

                document.getElementById("reussite").style.display = "block";

                document.getElementById("reussite-entete").innerHTML = `<h1>ENCORE UN
PEU<br>Votre score est ${score}</h1>`;

                document.getElementById("reussite-corps").innerHTML = ``;

                document.getElementById("reussite-empied").style.display = "none";

                setTimeout(() => {

                    document.getElementById("reussite").style.display = "none";

                    btnContinuer.click();

                }, 1750);

                actuel++;

            } else {

```

```

        //Écran de réussite de niveau

        actuel = 1;

        document.getElementById("reussite").style.display = "block";

        document.getElementById("reussite-entete").innerHTML = `<h1>BRAVO VOUS AVEZ
RÉUSSI LE NIVEAU ${niveau}</h1>`;

        document.getElementById("reussite-corps").innerHTML = `<h3>Votre score est
${score}<br>Votre meilleur score est ${JSON.parse(localStorage.getItem("Meilleurs-
Scores"))[0].Score}</h3>`;

        document.getElementById("reussite-enpied").style.display = "block";

        niveau++;

    }

    modifie_curseur("default");

    }, 1100);

    position = 0;

    btnDepart.disabled = false;

    jouer = false;

}

}

});

});

});

```

```

//-----ÉCOUTEURS D'ÉVÉNEMENTS POUR COMMENCER LE JEU-----\\

btnDepart.addEventListener("click", ()=>{

    score = 0;

    modifie_curseur("not-allowed");

    btnDepart.style.display = "none";

    //Fait clignoter tous les boutons en ordre

    //for(let i = 0; i<tuiles.length; i++){setTimeout(() => clickBtn(tuiles[i],1000), i * 1000);}

    boutonsDeJeu = tableauHasard(niveau+2, parseInt(nombre.innerText));

```

```

//Fait clignoter tous les boutons selon l'ordre de la séquence

for(let i = 0; i < boutonsDeJeu.length; i++){setTimeout(() => clickBtn(tuiles[boutonsDeJeu[i]], 1000), i *
1100);}

setTimeout(() =>{
    jouer = true;
    modifie_curseur("pointer");
    }, boutonsDeJeu.length * 1000);

});

//-----PASSER AU NIVEAU PROCHAIN-----\\
/**
 * Le curseur ne permet plus de cliquer
 * L'arrière plan change de couleur selon la progression
 * On efface la fenêtre de réussite
 * On régénère les boutonsDeJeu
 * Et on les fait clignoter une fois
 * Après un temps calculé (1,2 secondes par boutonsDeJeu),
 * On permet le joueur de jouer
 */
btnContinuer.addEventListener("click", () => {
    modifie_curseur("not-allowed");
    //CouleurAP est le tableau qui contient tous les arrières plans
    document.querySelector(":root").style.background = CouleurAP[niveau%7];
    document.getElementById("reussite").style.display = "none";
    boutonsDeJeu = tableauHasard(niveau+2, parseInt(nombre.innerText));
    for(let i = 0; i < boutonsDeJeu.length; i++){setTimeout(() => clickBtn(tuiles[boutonsDeJeu[i]], 1100), i *
1200);}

    setTimeout(() =>{

```

```

        jouer = true;

        modifie_curseur("pointer");
    }, boutonsDeJeu.length * 1200);

});

//-----RECOMMENCER UNE PARTIE PERDUE-----\\
/**
 * On efface la fenêtre d'échec
 * On fait apparaître le bouton de départ
 * On remet l'arrière-plan au bleu-ciel
 */
btnRecommencer.addEventListener("click",() => {
    document.getElementById("echec").style.display = "none";
    btnDepart.style.display = "block";
    document.querySelector(":root").style.background = CouleurAP[niveau%7];
});

//-----PRÉVENIR L'ABANDONNEMENT D'UN PARTIE INCOMPLÈTE (selon MDN)-----\\
window.addEventListener("beforeunload", (e) =>{
    e.preventDefault();
    e.returnValue="";
});

```