

Création des variables utiles pour notre scripte : startGame, diceNumber, global1, global2, round, playerTurn, winScore

Pour le bouton New Game ➔

Créer la variable newgame en récupérant son id dans le dom

Créer la fonction NewGame qui récupère tout les id de score en les paramétrant sur 0

Récupérer l'id résultat-dé et lui afficher le dé n°1

Récupérer l'id du tour-joueur1 et lui afficher le style d'écran inline (affiche le point)

Récupérer l'id du tour-joueur2 et lui supprimer le style d'écran

Définition des valeurs des variables créer au-dessus

startGame = true (permet au bouton RollDice de se lancer uniquement si la partie est start)

les autres variables à 0 hormis le playerTurn à 1 et le winScore à 100

ajouter l'événement à la variable newgame par un click en prenant en compte la fonction NewGame

Pour le bouton RollDice ➔

Créer la variable « rolldice » en récupérant l'id correspondant sur la page html

Créer la fonction RollDice

Si (startGame) {

Var dice = créer la formule permettant de générer un nombre aléatoire entre 1 et 6 ;

Var diceImg = récupérer les images de la classe .dice ;

La source de la variable diceImg = dossier image + var dice + extension ;

Si le dé n'est pas égale à 1

Ajouter à round le résultat de dice ;

récuper l'id score-temp-joueur, + playerTurn et ainsi l'ajouter au round

Sinon passer au joueur suivant

}

Ajouter l'évènement à la variable rollDice par un click et prenant en compte la fonction créer précédemment

Pour le bouton Hold ➡

Créer la variable « Hold » en récupérant son id sur la page HTML

Créer la fonction Hold

Si le tour du joueur est strictement égal à 1 {

Ajouter au score global 1 la variable round ;

Récupérer l'id score-joueur ainsi que PlayerTurn et ainsi l'ajouter au global1 ;

Ajouter la fonction Win en dernier qui détectera si oui non le score à atteint les 100

Sinon

Faire la même manipulation mais en prenant global2 }

Ajouter la fonction Win en dernier qui détectera si oui non le score à atteint les 100

Ajouter l'évènement à la variable hold par un click et prenant en compte la fonction Hold

Pour passer au joueur suivant ➡

Créer la fonction NextPlayer {

Si le tour du joueur est strictement égal à 1 {

L'autre tour de joueur est égal à 2

Son round est donc de 0

Récupérer l'id 'score-temp-joueur1 et le passer à 0 ;

Récupérer l'id 'tour-joueur1' et lui enlever les styles à l'écran (supprime le point indiquant que c'est son tour);

Récupérer l'id 'tour-joueur2' et lui ajouter un style à l'écran inline (affiche le point indiquant que c'est son tour ;

Sinon

```
{ faire la même manipulation mais en passant le playerTurn à 1 et en inversant les récupérations d'id  
  
}
```

Ajouter la fonction alertNextPlayer permettant d'afficher un message pour chaque changement de joueur

La fonction Win ➔

Créer la fonction Win () {

Si le score global1 est supérieur ou égal au winScore ou si le score global2 est supérieur ou égal au WinScore {

Ajouter la fonction alertWin permettant de dire quel est le joueur qui a gagné ainsi qu'un message demandant si nous voulons faire une autre partie ;

Ajouter la fonction NewGame permettant de créer une nouvelle en réinitialisant les données

Sinon

Ajouter la fonction NextPlayer

}

Création des alertes ➔

Créer la fonction alertNewGame

Création d'une Sweetalert(provient de bootstrap) nous indiquant que nous allons commencer une nouvelle partie ;

Créer la fonction alertNextPlayer

Création d'une Sweetalert nous indiquant que c'est au joueur suivant ;

Créer la fonction alertWin

Création d'une sweetalert nous indiquant que c'est le joueur + tourJoueur qui a gagné et qui signale qu'une nouvelle partie va commencer