myapp

+ names : var
+ pts : var
+ players : var
+ numPlayers : var
+ lowest_score : var
+ prevLowScore : let
+ playerWinning : let
+ playersPlayed : let
+ prevLowScore : let

+ io.on('connection', function(socket))
+ socket.on('set_username', function(username))
+ socket.on('player_Points', function(playerPoints))
+ socket.on('disp_scoreToBeat',function(scoreToBeat))
+ socket.on('check_Winner',function())
+ socket.on("disconnect", function())

dice

+ dices : let

+ constructor(diceID)
+ console(res)
+ stop()
+ random()
+ print(res)
+ processing()
+ cycle()
+ throw()
+ delete()

index

+ socket : var
+ username : var
+ container : let
+ throwBtn : let
+ playerPoints : let
+ scoreToBeat : let
+ diceCycleFor : let
+ element : var
+ diceCycleFor : let

+ showUsername()
+ showGame()
+ socket.on('username_set', function(msg))
+ socket.on('log', function(msg))
+ socket.on('game', function(msg))
+ socket.on("disconnect", function())
+ socket.on('waiting_for_players', function(numPlayers))
+ socket.on('begin_Game',function())
+ socket.on('winner', function(playerWinning))