

```

//Settings
var MaxProfitMode = true;
var PercentOfTotal = 0.001;    // here set percent for your initial bet of your
total balance
var BaseBet = 2;
var Multiplier = 1.25;
var dalembert = 1;
var MaxBet = 10000;
var MaxProfit = 50000000;
var MaxLoss = 10000;
var RandomBreak = 0;

var Username = engine.getUsername();
var StartBalance = engine.getBalance();
var CurrentGameID = -1;
var FirstGame = true;
var CurrentBet = BaseBet;
var CurrentMultiplier = Multiplier;
var d = new Date();
var StartTime = d.getTime();
var LastResult = "WON";
var Break = false;
var StartBet = BaseBet;
// Check previous bet
var LastBet = 0;
var LastProfit = 0;
var NewProfit = 0;
// Recovery variables
var SessionLost = 0;
var last1 = 1;

// Welcome message
console.log('%c[Startup]' + '%c Start balance is: ' + (StartBalance /
100).toFixed(2) + ' bits', 'color:orange;', 'color:black;');

//check if the multiplier is 1 or higher.
if(Multiplier < 1){
    console.error('[Bot] Your multiplier must be 1x or higher');
    engine.stop();
}

// Start of a game.
engine.on('game_starting', function(info) {
    CurrentGameID = info.game_id;
    console.log('-----');
    console.log('%c[Bot]' + '%c Game #' + CurrentGameID + ' started',
'color:orange;', 'color:black;');

    var random = randomNumber(1,100);

    if(last1 < 0){
        console.log('%c[Bot]' + '%c Break this round', 'color:orange;',
'color:black;');
        Unit = 1;
        Break = true;
    }
}

```

```

    if(Break === false){
        if(MaxProfitMode === true){
            BaseBet = Math.round((PercentOfTotal / 100) * (engine.getBalance() /
100).toFixed(2));
        }

        if (LastResult == 'LOST' && !FirstGame) { // Check if you lost the last
game
            CurrentBet = LastBet * 4;
            CurrentMultiplier = Multiplier;
        }

        //check if current bet is 0 or negative
        if(CurrentBet < 1){
            CurrentBet = 1;
        }

        // First game is set to false.
        FirstGame = false;
        // Changing last result
        if(((engine.getBalance() / 100) - CurrentBet) < ((StartBalance / 100) -
MaxLoss)){
            console.log('%c[Bot]' + '%c This bet would exceed Your maximum loss,
the bot will stop now... ', 'color:orange;', 'color:red;');
            engine.stop();
        }else{
            if (CurrentBet <= engine.getBalance()) { // Check if the balance is
high enough to place the bet.
                if (CurrentBet > (MaxBet)) { // Check if the bet is higher than the
given maximum bet by the user.
                    console.warn('[Bot] Current bet exceeds your maximum bet. Your
bet is changed to: ' + (MaxBet) + ' bits');
                    CurrentBet = MaxBet;
                }
                console.log('%c[Bot]' + '%c Betting ' + (CurrentBet) + ' bits,
cashing out at ' + CurrentMultiplier + 'x', 'color:orange;', 'color:blue;');
                engine.placeBet(CurrentBet * 100, Math.round(CurrentMultiplier *
100), false);
                LastBet = CurrentBet;
                LastProfit = (CurrentBet * CurrentMultiplier) - CurrentBet;
            }
            else { // Not enough balance to place the bet.
                if (engine.getBalance() < 100) { // Stop the bot if balance is less
then 100 bits.
                    console.error('[Bot] Your account balance is too low to place a
bet.... the bot will close now');
                    engine.stop();
                }
                else { // Changes basebet to 1 if balance is to low to make the
current bet.
                    console.warn('[Bot] Your balance is too low to bet: ' +
(CurrentBet / 100) + ' bits. ');
                    BaseBet = 1;
                }
            }
        }
    }
}
else{

```

```

        Break = false;
    }
    if (LastResult == 'WON') {
        CurrentBet = BaseBet;
        CurrentMultiplier = Multiplier;
    }
});

engine.on('cashed_out', function(data) {
    if (data.username == engine.getUsername()) {
        console.log('%c[Bot]' + '%c Successfully cashed out at ' + (data.stopped_at /
100) + 'x', 'color:orange;', 'color:green;');
        SessionProfit = SessionProfit + (Unit * MaxSessionProfit);
        if(((engine.getBalance() - StartBalance) / 100).toFixed(2) > MaxProfit){
            console.log('%c[Bot]' + '%c Maximum profit reached, bot is shutting
down...', 'color:orange;', 'color:green;');
            console.log('%c[Bot]' + '%c You have made ' + ((engine.getBalance() -
StartBalance) / 100).toFixed(2) + ' profit this session', 'color:orange;',
'color:green;');
            engine.stop();
        }
        LastResult = "WON";
    }
});

engine.on('game_crash', function(data) {
    var newdate = new Date();
    var timeplaying = ((newdate.getTime() - StartTime) / 1000) / 60;
    console.log('%c[Stats]' + '%c Game crashed at ' + (data.game_crash / 100) +
'x', 'color:orange;', 'color:black;');
    if(engine.getBalance() - StartBalance >= 0){
        console.log('%c[Stats]' + '%c Session profit: ' + ((engine.getBalance() -
StartBalance) / 100).toFixed(2) + ' bits in ' + Math.round(timeplaying) + '
minutes', 'color:orange;', 'color:green;');
    } else if(engine.getBalance() - StartBalance < 0) {
        console.log('%c[Stats]' + '%c Session profit: ' + ((engine.getBalance() -
StartBalance) / 100).toFixed(2) + ' bits in ' + Math.round(timeplaying) + '
minutes', 'color:orange;', 'color:red;');
    }
    if (data.game_crash >= CurrentMultiplier) {
        LastResult = "WON";
    }else{
        LastResult = "LOST";
    }
});

function randomNumber(min,max)
{
    return Math.floor(Math.random()*(max-min+1)+min);
}

```