**RTP adjust option**

Due to numerous requests from customers, we have added an optional feature to adjust slot machine payouts (for testing purposes).

|  |
| --- |
| How does this work:  - the script defines all possible positions of the slot machine reels. An array of slot machine windows is created.  - for each slot machine window, the money won, free spins, jackpots are calculated (to speed up the calculations, we have added a multi-threaded mode). In this case, the current slot lines of the machine are taken into account.  - all calculation data is saved in files (json) so that they can be used in the finished game.  **Accordingly, each slot machine must have its own copy of calculation files. When making any changes to the slot machine files, you need to run the RTP calculator and update the calculation files.** |

For this purpose, there are 2 scripts in the asset: RTPCalc.cs, RTPAdj.cs. These scripts must be added to the SlotController object.

|  |
| --- |
| First, add the RTPClalc.cs script. Make duplicates of the calculation files (.json) and drag them into the corresponding fields of the script.  In this screenshot you can see a duplicate of scene number 2 with the payout adjuster added. |

|  |
| --- |
| To update the payout calculation files, click the <**Calc winsTh**> button and wait for the work to be completed. This may take a long time. The execution time depends on the number of symbols on the slot reels, the number of slot lines, the length of the payout table ...    You can also increase the number of threads to speed up the calculations. On my PC, the script works well on 16 threads (AMD Ryzen 7 4000 series).  Max Lines Count - here you can limit the number of slot lines only for the test. But for a full calculation you need to specify the full number of slot lines. If you specify more lines, no problem, the script itself will determine the correct number of lines. |

|  |  |
| --- | --- |
| I do not recommend using the single-thread mode, the script execution time is very long. |  |

|  |
| --- |
| The wins.json file contains absolutely all winning combinations, including scatter wins, free spins, and jackpots.  (winFS.json – only free spins wins; winsJP.json – only jackpot wins; winsScatter.json – only scatter wins) |

|  |  |
| --- | --- |
| After executing the script in the editor console you can see: the number of calculated slot reel combinations, the number of wins ... |  |

|  |
| --- |
| Changes in the SlotController.cs script:  here the random data of the random number generator can be replaced with data from the payout adjuster. If the component is added to the object and enabled, we will receive new data for the slot machine reels. |

Add RTPAdj.cs component to the SlotController game object and choose <**AdjMode**>.

Operating modes of the payment adjuster (RTPAdj.cs) :

|  |  |  |
| --- | --- | --- |
|  | FULLRANDOM | The slot controller will continue to operate with completely random values. |
| SCATTERWINS | A random position from the winsScatter.json file will be returned. |
| WINS | A random position from the wins.json file will be returned. |
| LOSSES | A random position from the losses.json file will be returned. |
| JPWINS | A random position from the wins.json file will be returned. |
| AUTO | The specified RTP will be automatically maintained. Of course, deviations are possible. |

|  |  |
| --- | --- |
| The payout adjustment algorithm is very simple. If the payout is less than the set one, a winning combination is issued; if the payout is greater than the set one, a losing combination is issued. |  |

|  |
| --- |
| Limitations:  The RTP adjuster will not work correctly with the HOLD functionality and Manual Stop of the slot. |