

UdonChips 2.0

開発者向けマニュアル

はじめに

このマニュアルは、UdonSharp を書いてギミックを作成するユーザー向けです。

ギミックを作らない、UdonSharpを利用しないユーザーは、別途「UdonChips ユーザー向けマニュアル」をご覧ください。

[UdonChipsユーザー向けマニュアル](#)

概要

UdonChipsは、ワールド内の「所持金」の共通規格です。

以下の機能は、一般ユーザーが変更する範囲です。



- **float money** : 所持金。
- **float initialMoney** : 初期所持金。
- **string format** : 所持金表示の共通フォーマット。表示単位の入れ替え。(¥、\$など)
- **SaveMode saveMode** : セーブを利用するかどうか。

以下の機能は、ワールドクリエイター向けです。

- **Offline**: オフライン中の所持金の変動を管理。
- **Auto Save Interval Seconds**:
saveMode が **Auto** のとき、**指定秒数ごとに自動セーブをチェック**します。
最後にセーブしてから所持金に変更があった場合のみ、セーブが実行されます。
極端に短いとセーブデータ更新のスパムにより、負荷が発生する可能性があります。
- **saveDataKeyMoney** :
PlayerDataに「money」を保存するキー値。
- **saveDataKeyLastSaveTime**:
PlayerDataに「最後にセーブした日時(DateTime)」を保存するキー値。

ギミック開発ルール: 表示フォーマット

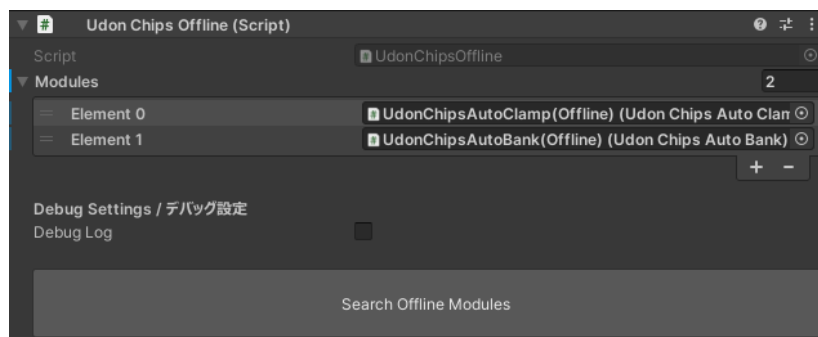
各ギミックは、ユーザーが所持金の「単位」を任意に変更できるようにするため、所持金の表示をするときは、以下の疑似コードのように、UdonChips.format を利用することを推奨します。

Format	{0:F0}uc
--------	----------

```
Text text;  
UdonChips udonChips;
```

```
void Update()  
{  
    text.text = string.Format(udonChips.format, udonChips.money);  
}
```

ギミック開発ルール:オフライン対応



オフライン中の時間や、日数に応じて、ログイン時に何らかの変更を加えたい場合、**UdonChipsOfflineModuleBase** (以下、OfflineModule) クラスを継承してください。

- **public virtual void OnPostLoadUdonChips(UdonChipsOffline offline)**
ログイン直後、UdonChipsでPlayerDataの読み込み完了直後に呼ばれます。
- offline からは、以下のデータが確認できます。
- loadedMoney : ロード直後の所持金
- savedTime : 前回の最終セーブ時刻
- loadedTime : 今回のロード時刻

初回ロード時は、savedTime == loadedTime になります。

必要に応じて、loadedMoney に変更した所持金を書き戻してください。
この関数内でUdonChips.moneyを参照しないでください。ロード前の不正な値です。

- **public virtual void OnCompleteLoadUdonChips(UdonChipsOffline offline)**
ログイン直後の、すべてのOfflineModuleの効果が適応され、最終的な所持金が確定した後に呼び出されます。
ロード直後のUI更新を想定しています。
- **public virtual void OnPreSaveUdonChips(UdonChipsOffline offline)**
セーブされる直前に金額を変更できます。
必要に応じて、loadedMoney に変更した所持金を書き戻してください。
この関数内でUdonChips.moneyに書き戻さないでください。
- **public virtual void OnPreSaveUdonChips(UdonChipsOffline offline)**
セーブが終わった直後に金額を変更できます。
- 必要に応じて、loadedMoney に変更した所持金を書き戻してください。
この関数内でUdonChips.moneyに書き戻さないでください。