オフライン環境用 難易度表作成用テンプレート ＆ サンプルセット の利用マニュアル　（2026/11 v2.0）

コラム -難易度表をオフライン（ローカル）環境で作成・登録する方法について-
https://pmsdifficulty.xxxxxxxx.jp/column_local_table.html
https://pmsdatabase.github.io/column_local_table.html（ミラーサイト）

表計算ソフト（.xlsx / Excel 等）でデータを作成し、URL の代わりにファイルパスから難易度表を登録することで、オフライン環境でも難易度表を楽しめるようにするためのサンプルセットです。

本マニュアルでは、オフラインユーザー向けにサンプルセットの使用方法をまとめます。
2026/11 beatoraja 0.8.9 アップデートに伴う重要な仕様変更に伴い、改訂を行いました。**以降の内容は、beatoraja 0.8.9 を前提とします。**それ以外のバージョンを使用する場合は、マニュアル最下部の Tips を確認してください。

解説する内容
- 表計算ソフトで難易度表と段位認定のデータを作成する方法
- ローカル環境にある難易度表データを LR2 / beatoraja で読み込ませる方法

関連するソフトウェア等
- メモ帳アプリ：Windows 付属でも何でも OK。参考画像は Terapad を使用したものになります。（必須）
- 表計算ソフト（.xlsx に対応しているもの。Excel 2019 で動作確認しました。必須）
- BeMusicSeeker（LR2 で必須）　※本マニュアルでは BeMusicSeeker Unofficial Fork で動作確認しました
  https://neeted.github.io/bemusicseeker-unofficial-fork/index.ja.html
  https://github.com/SayakaIsBaka/BeMusicSeeker-installer/releases/tag/installer-v1
- DB Browser for SQLite : https://sqlitebrowser.org/（難易度表用データの取得時にあると便利）

- Windows 10,11 での操作に基づきます。その他 OS では適時読み替えてください。
- ファイルの拡張子を表示： エクスプローラ ＞ 「表示」タブ ＞ 「ファイル名拡張子」をチェック
- 文字コードを変換： メモ帳 ＞ ファイル ＞ 名前を付けて保存 ＞ 「文字コード」プルダウンメニュー

サンプルセットの中身の概要
- **worksheet.xlsx** … ここでデータ作成を行います。**.xlsx** に対応した表計算ソフトが必要です。
- **header.json** … 難易度表全般のプロパティと段位認定情報を書き込むファイルです。
- **score.json** … 難易度表に登録する譜面の情報を書き込むファイルです。
- **table.html** … beatoraja 0.8.9 以降で、ローカル難易度表を読み込ませる際に使用します。

- **header_with_course_sample.json / course_sample.lr2crs** … 解説用のサンプルファイルです。
- **manual.pdf** … このマニュアルです。

表計算ソフトで難易度表のデータを作成する

(1) header.json の編集

難易度表全般のプロパティが格納されているのが「header.json」ファイルです。
メモ帳で「header.json」を開き、次の項目を更新してください。後日変更することも可能です。

■ **"name" : "難易度表の名前を記入"**
■ **"symbol" : "シンボルを記入"**　*★とか◆とかのアレ。オンライン公開しない場合は適当に決めて OK
■ **"data_url" : "score.json のある URL を記入"**　*初期状態 **./score.json** のままで編集不要
■ **"level_order" : [難易度を記入]**　*カッコ内に使用する難易度をカンマ区切りで全て記入。数字以外は""で囲む。

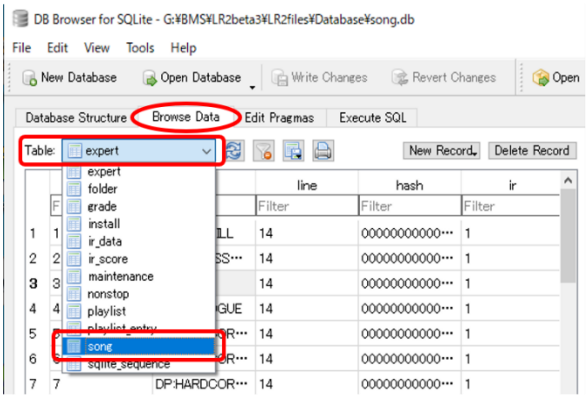
(2) score.json の編集

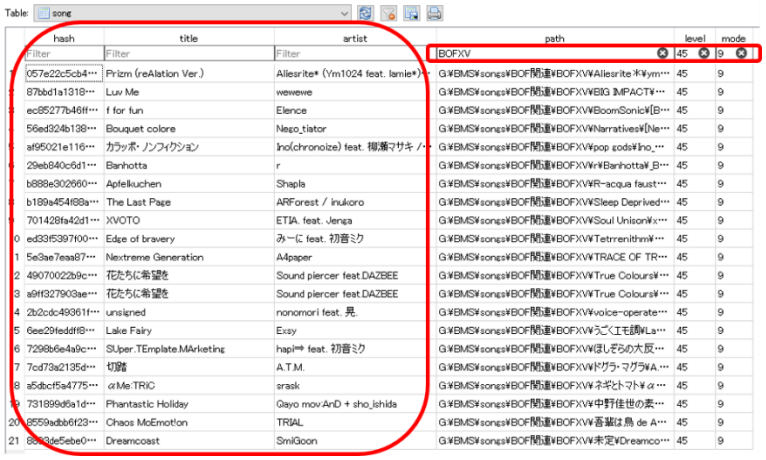
難易度表に登録する譜面の情報が格納されているのが「score.json」ファイルです。
このファイルに書き込むデータを表計算シートで作成していきますが、**シートでは セル内改行 および セルの切り取りは行わないでください。(コピペ上書きは OK)**
「worksheet.xlsx」の「テンプレート（難易度表)」シートを開き、譜面についての次の情報を記入してください。

■ **難易度** …「header.json」で指定した数字と記号に対応させる。記号の半角/全角入力ミスに注意
■ **曲名**
■ **md5 (or LR2IR)** … 譜面の識別番号のようなもので、様々な取得方法があります。
・beatoraja の選曲画面で当該曲にカーソルを合わせて Ctrl + F3 でコピーされる 32 桁の文字列(md5) を記入
・Table Edidor ツール や BMSHashUtil に BMS ファイルを投入し、出力される md5 を記入
Table Edidor : http://stairway.sakura.ne.jp/bms/tabletool/editor.html
BMSHashUtil : https://mistyblue.info/tools.html

・DB Browser for SQLite を利用してまとめて取得
DB Browser for SQLite : https://sqlitebrowser.org/
こちらはオフライン環境では特にオススメの方法です。下記ディレクトリにある拡張子.db のファイルをこのアプリで開くことで、md5、曲名、レベルなどの曲情報をまとめて取得し、ワークシートにコピペすることができます。
使用前に BMS プレイヤー側で全楽曲データの読み込みを行っておきましょう。
LR2 の場合:　LR2beta3 /LR2files /Database /song.db
beatoraja の場合:　beatoraja ディレクトリ内の songdata.db

DB Browser for SQLite を開いて DB ファイルを D&D で投入し、Browse Data > Table: のプルダウンメニューから"song"を選択します。読み込み済の曲情報が表形式で表示されるので、必要な情報をコピーしましょう。song.db では表カラムの"hash" が md5 に相当しています。
フィルター機能を利用すると効率が上がります。例えば「BOFXV フォルダ内にあるレベル 45 表記の PMS(9keys)」を難易度表にしたい場合、"path" に「BOFXV」"level"に「45」"mode"に「9」を入力し、抽出された譜面リストから"hash" "title" "artist" などをコピーし、ワークシートに貼り付ければ OK です。





■ **sha256 (or Mocha-Repository)** … BMSON 譜面を難易度表に追加する場合、md5 の代わりに記入します。

・ beatoraja の選曲画面で Ctrl + Shift + F3 キーを押した際にコピーされる文字列を貼り付ける

・ DB Browser for SQLite で songdata.db (beatoraja) を開き、カラム “sha256” から取得

■ **その他の項目(youtube ID, 曲作者, コメント等)** … オフラインでのみ使用する場合は入力不要です。

ただし、列を削除すると動作不良を起こしますので、見づらい場合は非表示化で対応してください。

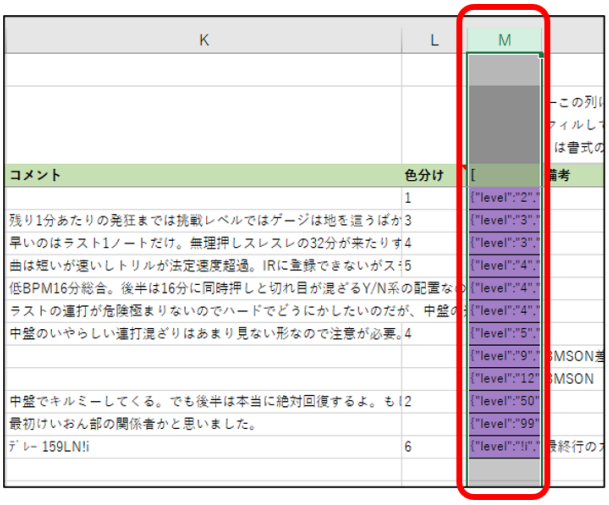
必要な情報を入力すると、M 列 紫色になっているセルにコードが出力されます。出力が足りない場合は、ドラッグフィル（コード出力済のセルを選択 > 右下の四角マークを下方向にドラッグ）で増やします。

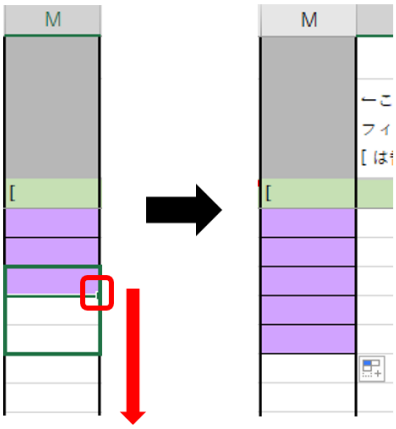
このコードを M 列全体でコピーし、「score.json」をメモ帳アプリで開いて貼り付けます。

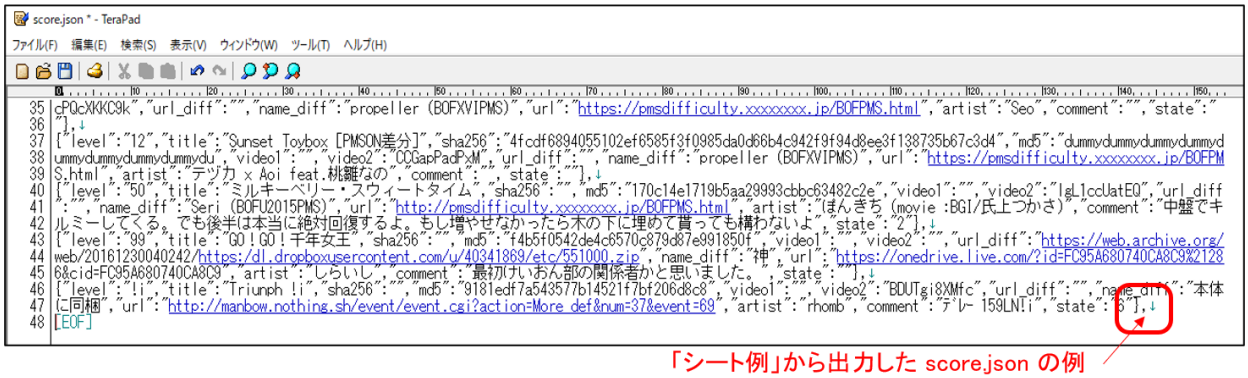
最後に、コードの最終行の末尾 }, を手作業で }] に書き換えてから保存 すれば完成となります。

文章での説明では直感的に理解し辛いと思いますので、サンプルを用意しています。

初期状態の「score.json」の中身は、ワークシート「シート例（難易度表)」のコードを列全体コピペし、末尾の }, を }] に変更したものです。「シート例」と「score.json」の中身を照らし合わせて確認してみてください。







(Tips)「score.json」作成時の注意点まとめ

・ セル内改行、セルの切り取りが使われていないかを確認する。セルの切り取りを誤って使用してしまった場合は、M 列の数式をテンプレートから再取得し、上書きし直します。

・ md5 または sha256 が入力されているか。例外として、旧 LR2IR および Mocha-Repository の曲別 IR ページへの URL については、URL をそのまま貼り付けても動作するように設計しています。

・ score.json 末尾の }, を }] に書き換えたか確認する。更新の度に行ってください。

・ ワークシートの列がみだりに削除されていないか。コードの出力が M 列以外になっている場合は要確認です。

・ BMSON の登録時は md5 ではなく sha256 を記入する。

・ 通常は問題になりませんが、新たに JSON ファイルを作成する場合は **文字コード UTF-8** を使用してください。

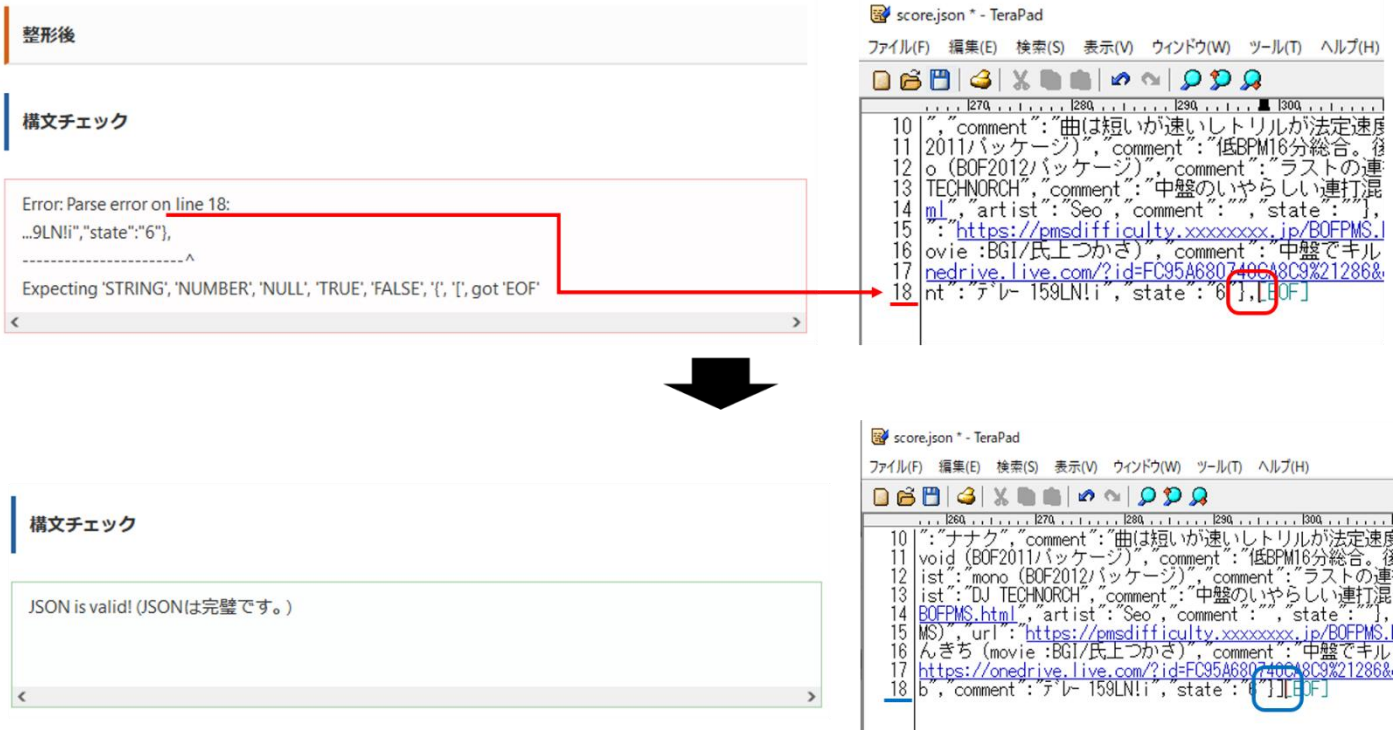
(Tips) JSON ファイルにエラーがないかどうかを確認する

JSON Pretty Linter Ver3 : https://lab.syncer.jp/Tool/JSON-Viewer/

こちらの Web ツールの「整形前」に .json ファイルのコードを貼り付けると、構文上のエラーが検出されます。
セル内改行や }, → }] 変換忘れは見落としやすいミスですので、何らかの原因で難易度表が読み込めなくなってしまった場合はお試しください。

エラーコード内の line 番号はワークシートの行番号とも一致するため、原因箇所の特定に役立ちます。

「header.json」「score.json」をはじめ .json のファイル全般に有効です。広く試してみてください。



表計算ソフトで段位認定のデータを作成する

この作業では「header.json」ファイルを改変していくのですが、改変後の「header.json」に 1 コースも登録されていない場合、難易度表の読み込みに不具合が発生します。
段位認定が必要なければこの工程はスキップし、次項「ローカル環境にある難易度表データを LR2 / beatoraja で読み込ませる」に進んでください。作業する場合は必ず 1 コース以上作成する所まで行いましょう。

(1)「テンプレート（段位認定)」を開き、B1～B4 セルに「header.json」の情報を書き写す
前段落で解説した「header.json」の入力手順に基づき、既存データを引用すれば OK です。
・ JSON 書式ミス防止のための処置です。JSON の仕組みを把握していて 既存データの体裁を変更したくないという場合は省略し、最終出力の "course" 要素のみ追加してください。
・ 難易度表をコース専用に使う場合も data_url は必ず指定しなければなりません。その場合「score.json」の中身は [] と入力(ブランク）しておけば大丈夫です。
・ B5 セル「header_md5」は初期状態のままにしておいてください。

(2) コース情報の記入
「シート例（段位認定)」を参考にしながら、コース情報を記入してください。**【必須】**となっている項目に記入漏れがあると、コースデータが出力されない仕様となっています。

■ コースタイトル（B 列)【必須】 *作成、テスト後に変更してもスコアデータは引き継がれますが、IR には一番最初にスコアを送信した時点でのコースタイトルが登録されます。
■ Line(LR2) (D 列)【必須】 … 使用するレーン数を半角数字で記入してください。
■ 1 曲目の曲情報（E～G 列)【必須】 … md5 が必須で、URL でも md5 文字列のみでも動作します。取得方法は前段落を参照してください。*難易度表に登録していない譜面でも使用できます。
■ 2～5 曲目の曲情報（H～S 列) … 4 曲構成以外も対応しています。必要な分だけ記入し、使わない所は空欄にしておいてください。

(3) カスタマイズ(beatoraja) (T～X 列) の記入
beatoraja で利用できる段位認定のオプション制限項目です。(トロフィー機能には対応しておりません。)
「シート例」や **次ページ「カスタマイズ項目の補足情報」**を参考に、文字列コピペで正確に書き写してください。
LR2 の段位認定に出来るだけ近づけたい場合は、「grade」欄に **grade_mirror** 「gauge」欄に **gauge_lr2** 「LN」欄に **ln** を指定し、**「HS」「judge」は空欄** にしてください。

(4) 全コースのデータ作成
(2)～(3) を繰り返し、全コースのデータを記入します。行数が足りない場合は黒線より上に行追加してください。
コースリストのソートも検討しておきましょう。作成したコースは、beatoraja では行番号昇順(シートの並びと同順)、LR2 では降順(シートの並びと逆順) で表示されます。

カスタマイズ項目の補足情報（厳密な仕様は調査しておらず、誤りがあるかもしれません）

■ grade

・**grade_mirror**： 譜面オプション MIRROR を許可

・**grade_random**： 譜面オプション RANDOM, R-RANDOM, S-RANDOM, MIRROR を許可

・**(空欄)**: grade_random と同様に扱われます。Mocha-Repository では通常の楽曲同様に使用 OP が記録されます。

・今回紹介している手法の場合、正規縛りは設定不可？ のようです。

■ HS

・**no_speed**： ハイスピード設定無効

・**(空欄)**： ハイスピード設定有効

■ judge

・**no_good** : GOOD 判定無効(BAD 判定になります)

・**no_great** : GREAT/GOOD 判定無効(BAD 判定になります)

・**(空欄)**： 全判定有効

■ gauge

・**gauge_lr2** : LR2 段位ゲージを使用

・**gauge_5k** : 5KEY 用段位ゲージを使用

・**gauge_7k** : 7KEY 用段位ゲージを使用

・**gauge_9k** : 9KEY 用段位ゲージを使用

・**gauge_24k** : 24KEY 用段位ゲージを使用

・**(空欄)**： 未検証

・異なるプレイスタイルのゲージ設定を適用する事も可能なようです。(9KEY 段位を 7KEY 段位ゲージで遊ぶなど)

・gauge_9k は gauge_lr2 よりも若干易しくなります。他のプレイスタイルは未検証ですが、恐らく独自のゲージ設定が備えられていると思われます。

・beatoraja の Practice モードでは段位ゲージを選択できますが、gauge_lr2 は使用不可？ のようです。

■ LN

・**ln**： 末端判定なし（LR2 仕様）

・**cn**： 末端判定あり（本家 CN）

・**hcn**： 押下中ゲージ増加、踏み外し中ゲージ減少（本家 HCN）

・**(空欄)**：LN/CN/HCN 全て選択可能で、Mocha-Repository では通常の楽曲同様に使用オプション毎のページ内リンクが作成されます。

(注意) beatoraja 0.8.4 で、grade および LN のカスタマイズが異なっていても、gauge 設定が共通していればスコアが共有されてしまう、という現象を確認しています。
例えば、beatoraja 上では次の 3 コースは同値となります。

- 曲目・曲順が一致し、カスタマイズが gauge_lr2 と grade_mirror / LN のコース
- 曲目・曲順が一致し、カスタマイズが gauge_lr2 と grade_mirror / **HCN** のコース
- 曲目・曲順が一致し、カスタマイズが gauge_lr2 と **grade_random** / LN のコース

HS, judge がついていれば別々に記録されます。
もしかしたら意図されていない挙動かもしれません。留意しておいてください。

(5) 段位認定データを登録する

LR2 の場合

ワークシートの Y 列(黄緑) に出力されたコードの列全体を、テキストファイルに貼り付けて保存します。

このとき文字コードは Shift-JIS で保存してください。

保存したファイルの拡張子を .lr2crs に変更すれば完成です。LR2SETUP 画面に D&D して確認しましょう。

一覧から削除したい場合は、LR2 の選曲画面 > TAG EDITOR > DELETE で行えます。

サンプルセット内の「course_sample.lr2crs」 が「シート例」からの出力になっています。





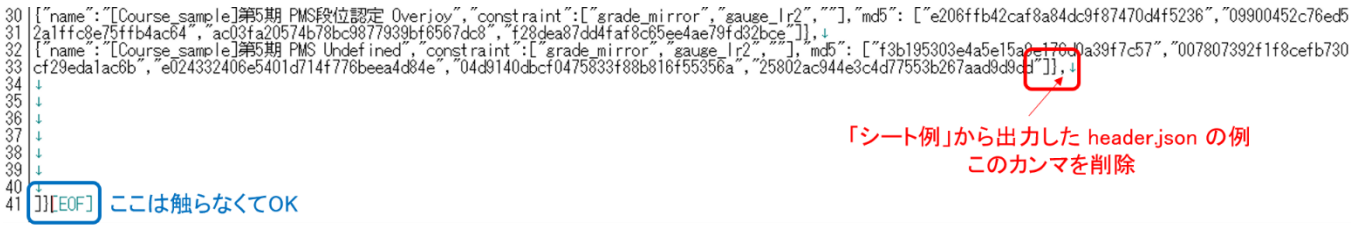
beatoraja の場合

ワークシートの Z 列(深緑) に出力されたコードの列全体を「header.json」に上書きします。

次に、コースデータ出力部の最終行 末尾 }], を手作業で }] に書き換えてから保存 すれば完成となります。

beatoraja に難易度表を読み込ませる(次項) ことで、フォルダ内から選曲可能です。

サンプルセットの「header_with_course_sample.json」が「シート例」からの出力となっています。



(Tips) コースデータ作成時の注意点まとめ

- 1 コースも作成しないまま「header.json」を上書きしていないか。
- 必須項目に記入漏れがないか、md5 が正確に記入されているか。
- セル内改行、セルの切り取りが使われていないか。列全体削除も使用不可です。
- LR2 用コースファイルの文字コードは Shift-JIS、「header.json」の文字コードは UTF-8 になっているか。
- header.json コースデータ出力部 末尾の }], を }] に書き換えたか確認する。更新の度に行ってください。
- BMSON は段位認定に組み込めません。
- 手違い等で B5 セル「header_md5」の値が変化すると、(恐らく) LR2 のコースファイルに不具合が生じます。

初期値のままに設定しておいてください。

ローカル環境にある難易度表データを LR2 / beatoraja で読み込ませる

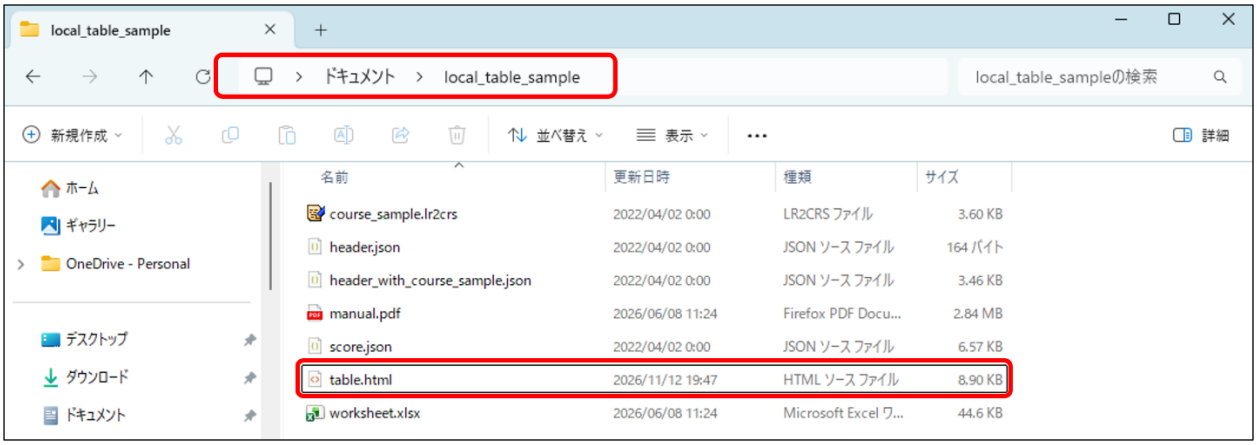
「table.html」「header.json」「score.json」が同一ディレクトリにあることを確認し、フォルダ全体の設置場所を確定してください。設置場所は自由に決めて構いませんが、ファイルパスに半角英数字(ASCII 文字) 以外が含まれないようにしてください。

次に、「table.html」の右クリックメニューからファイルパスを取得し、メモ帳アプリなどに貼り付けます。(Win10 以前の場合は ファイルを選択 ＞ Shift+右クリック ＞ パスのコピー)

貼り付けたパスからダブルクォーテーション(") を削除し、パス内の ¥ 記号を / に変換しておいてください。

以下は一例として、ユーザ名 "user" が C ドライブ ＞ ドキュメント にサンプルファイル「local_table_sample」全体を設置したものとして紹介します。

この場合のファイルパス　**C:/Users/user/Documents/local_table_sample/table.html**



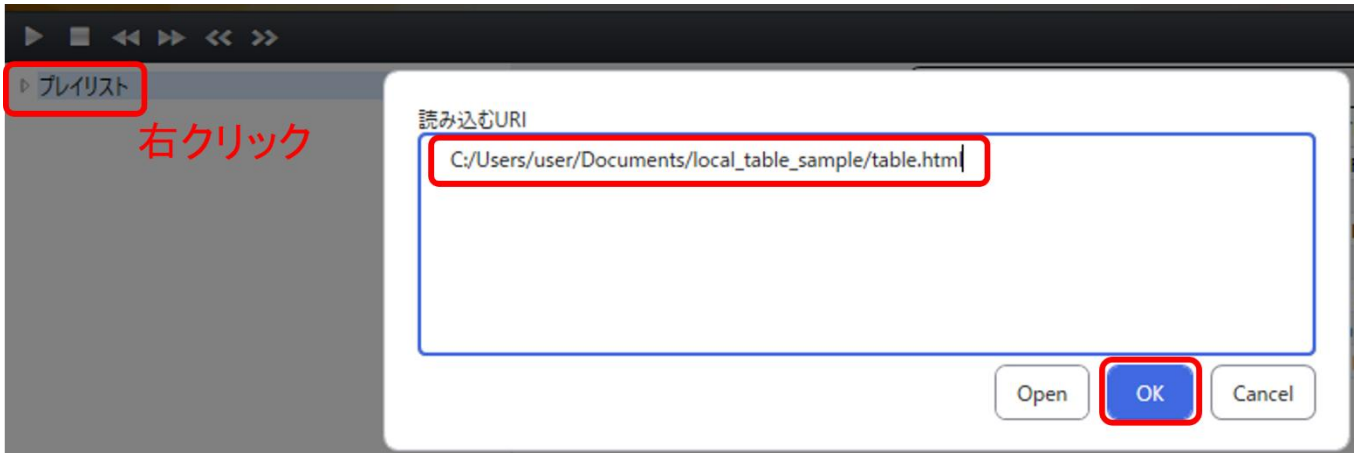
LR2 の場合（BeMusicSeeker）

BeMusicSeeker Unofficial Fork を使用します。あらかじめインストールしておいてください。Ribbit 氏の正規版や黒皇帝氏のミラー版 を既に使用している場合はそちらで構いません。

https://neeted.github.io/bemusicseeker-unofficial-fork/index.ja.html

https://github.com/SayakaIsBaka/BeMusicSeeker-installer/releases/tag/installer-v1

メイン画面の左側「プレイリスト」を右クリック ＞ インポート ＞ URL を指定して読み込む を選択し、上記のファイルパスを入力して OK を押すと難易度表として読み込まれます。



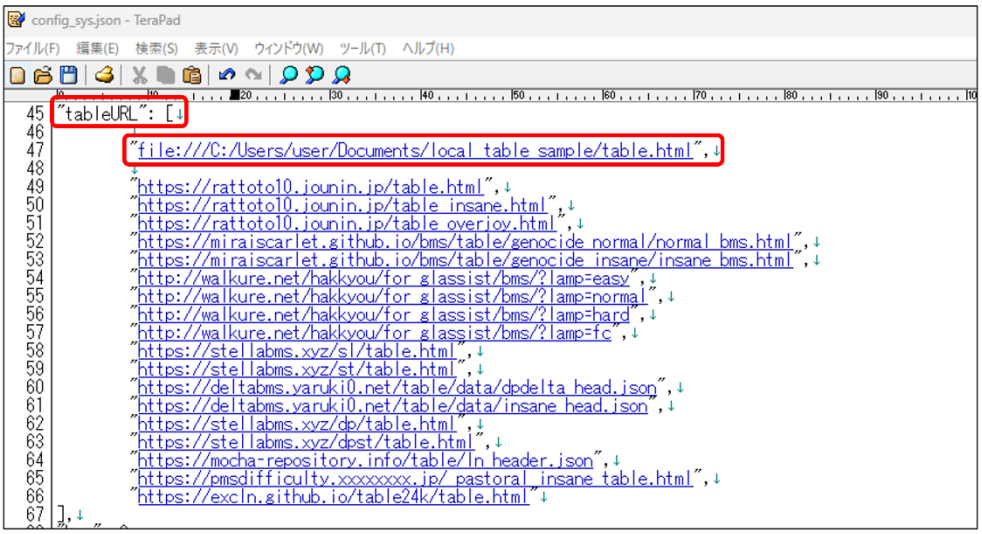
beatoraja の場合

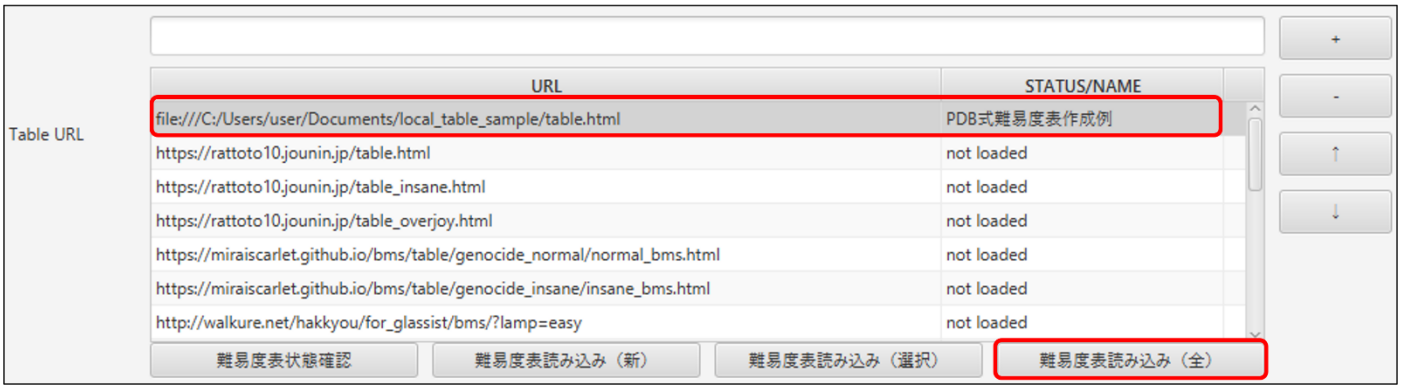
(beatoraja 0.8.9 現在) beatoraja-config > リソース > Table URL ではファイルパスが入力出来ませんが、URL の保存されている beatoraja/config_sys.json を直接編集する事で、ファイルパスから難易度表を読み込ませることが可能になります。ただし、**操作を誤ると config_sys.json が初期化されてしまうので、あらかじめバックアップを作成しておいてください。**

beatoraja フォルダ直下の config_sys.json をメモ帳で開き、**"tableURL":[** と書かれている行の下に新たに行を挿入し、次のように入力して保存します。

"file:///(table.html のファイルパス)",

上述の例の場合は **"file:///C:/Users/user/Documents/local_table_sample/table.html",**





きちんと導入できていれば、上図のように beatoraja-config > リソース > Table URL に難易度表が表示されています。「難易度表読み込み」を押して反映しましょう。

表示されない場合、先に一度リソースタブ内の「難易度表読み込み（全)」を実行してみてください。また、¥ 記号や非 ASCII 文字が含まれていないか、ファイルパス周りの記述にミスがないか、もう一度確認してください。

難易度表の更新等について

以降は「header.json」「score.json」を編集し、BeMusicSeeker あるいは beatoraja 上で更新すれば OK です。

難易度表のファイル一式を移動させたい場合は、いったん難易度表の登録を解除し、移動先のファイルパスで再度上記の導入作業を行ってください。2 つ目以降の難易度表を作成した場合も同様に行ってください。

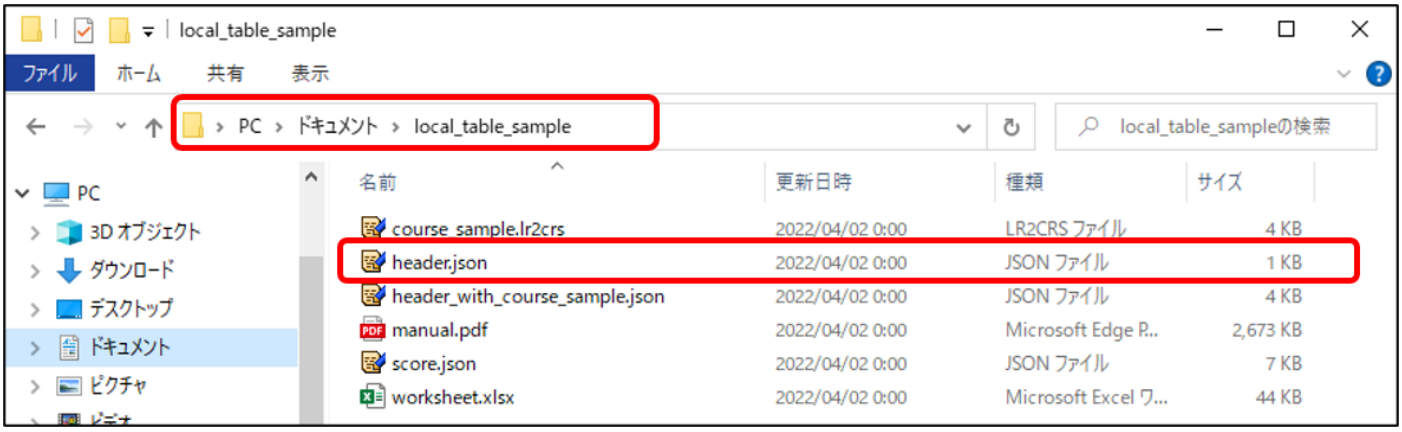
(Tips) beatoraja 0.8.8 以前のバージョンを使用している場合

beatoraja 0.8.6 ～ 0.8.8 の場合

table.html の代わりに header.json のファイルパスを取得し、config_sys.json で同様に記入します。

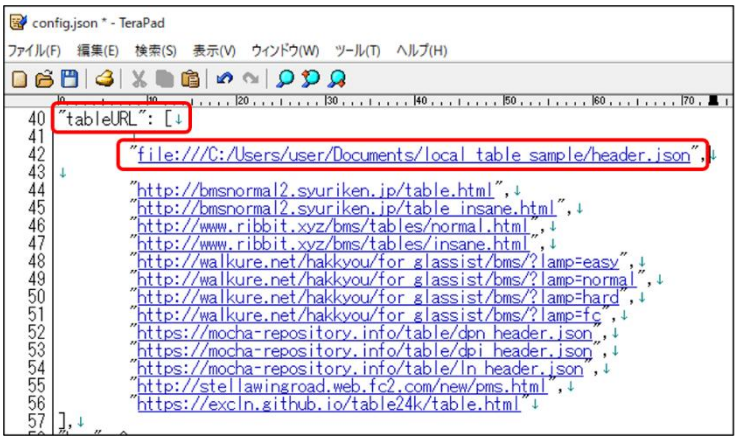
"file:///(header.json のファイルパス)",

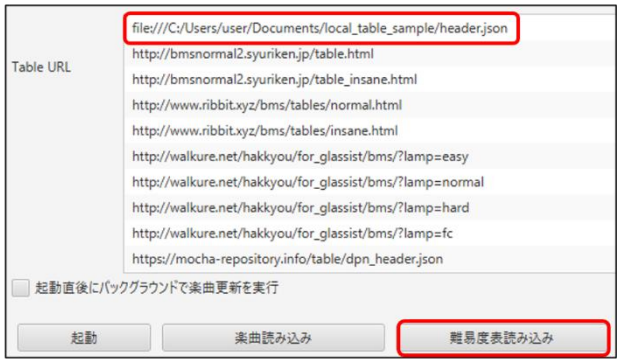
"file:///C:/Users/user/Documents/local_table_sample/header.json",



beatoraja 0.8.5 以前の場合

config_sys.json の代わりに config.json を開き、header.json のファイルパスを記入します。





(Tips) 他の方が公開している難易度表をローカル環境下にバックアップして使用する

Web 公開されている大半の難易度表は、ソースコード内の記述を介して header.json や score.json に直接アクセスできます。これらをダウンロードしておくと、難易度表の公開が停止された際のバックアップになり得ます。

<meta name="bmstable" content="(header.json の URL)" />

Insane Record - PMS Database (2026/11 現在) であれば、取得すべきファイルは次のようにアクセスします。
https://pmsdifficulty.xxxxxxxx.jp/insane_PMSdifficulty.html (難易度表の URL)
https://pmsdatabase.github.io/insane_pmsdatabase/insane_pmsdatabase_header.json (ソースコードから取得)
https://pmsdatabase.github.io/insane_pmsdatabase/insane_pmsdatabase_score.json (header.json から取得)

核シェルターでも難易度表 BMS ライフ！

更新履歴

2022/04 v1.0 公開
2026/06 v1.1 LR2IR の閉鎖に関連する解説の変更と、文章のマイナーチェンジ
2026/11 v2.0 マニュアル全体の刷新、beatoraja 0.8.9 の仕様変更に対応

連絡先
https://x.com/pmsdatabase
pvideocomplement@gmail.com

2027/06 以降、PMS Database は更新を終了し、アーカイブ状態に移行します。連絡を試みて頂いても構いませんが、お返事できない場合があります。昨今は AI エージェントが発達しており、そちらに質問する方がきっと有益です。
PMS Database の制作物に関しては、機械学習への利用・再配布などに制限は設けません。各自の責任で良い感じに改良してください！