

「#Json」 説明書

Jsonアニメーションファイル入出力ツール



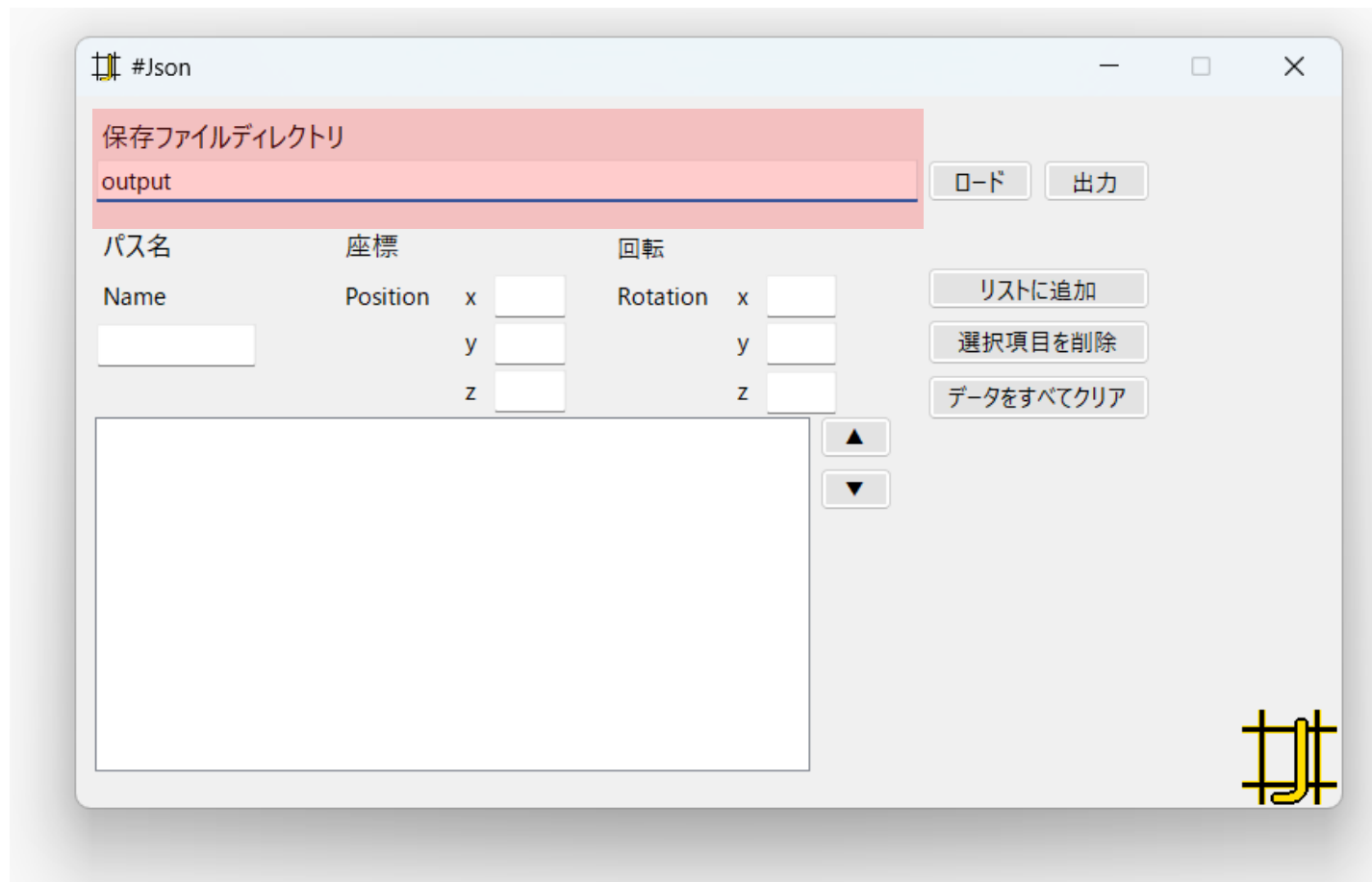
画面構成

The screenshot shows the #Json application window with the following components labeled with circled numbers:

- ① 保存ファイルディレクトリ (Save file directory): A text input field containing "output".
- ② ロードボタン (Load button): A button labeled "ロード".
- ③ 出力ボタン (Output button): A button labeled "出力".
- ④ パス名 (Path name): A text input field labeled "Name".
- ⑤ 座標 (Coordinates): A section with three input fields for "Position" (x, y, z).
- ⑥ 回転 (Rotation): A section with three input fields for "Rotation" (x, y, z).
- ⑦ リストに追加 (Add to list): A button labeled "リストに追加".
- ⑧ 選択項目を削除 (Delete selected item): A button labeled "選択項目を削除".
- ⑨ データをすべてクリア (Clear all data): A button labeled "データをすべてクリア".
- ⑩ 出力データリスト・順番移動ボタン (Output data list / Order movement button): A large empty list area with up and down arrow buttons on the right side.

- ①保存ファイルディレクトリ
- ②ロードボタン
- ③出力ボタン
- ④パス名
- ⑤座標パラメータ
- ⑥回転パラメータ
- ⑦リストに追加ボタン
- ⑧選択項目を削除ボタン
- ⑨データをすべてクリアボタン
- ⑩出力データリスト・順番移動ボタン

①保存ファイルディレクトリ



デフォルトではoutputディレクトリが設定されている。

※フォルダ内構成

output

#Json.dll

#Json.exe

#Json.runtimeconfig.json

Newtonsoft.Json.dll

出力先を変えたい場合は好きなパスに変更する。

②ロードボタン

The screenshot shows the #Json application window. At the top, there's a title bar with the icon and text '#Json'. Below it, a text field labeled '保存ファイルディレクトリ' (Save file directory) contains the text 'output'. To the right of this field are two buttons: 'ロード' (Load) and '出力' (Output). Below the directory field, there's a table with columns for 'パス名' (Path name), '座標' (Coordinates), and '回転' (Rotation). The 'パス名' column has a sub-column 'Name' with an input field. The '座標' column has sub-columns 'Position' and 'x', 'y', 'z'. The '回転' column has sub-columns 'Rotation' and 'x', 'y', 'z'. Below the table is a large empty text area. To the right of the text area are three buttons: 'リストに追加' (Add to list), '選択項目を削除' (Delete selected items), and 'データをすべてクリア' (Clear all data). At the bottom right of the window is a yellow icon.

パス名	座標	回転
Name	Position	Rotation
	x	x
	y	y
	z	z

クリックするとエクスプローラーが開かれる。

拡張子「.json」のファイルを選択して開くと、データリストに追加される。

【注意事項】

1. 読み込むたびに上書きされるため、消したくないデータは書き出しておく。
2. 「#Json」で書き出されたデータのみに対応しているため、他のデータは読み込めない。

③出力ボタン

The screenshot shows the #Json application window. At the top, there's a title bar with the icon and text "#Json". Below it, a "保存ファイルディレクトリ" (Save file directory) field contains "output". To the right of this field are "ロード" (Load) and "出力" (Output) buttons. Below the directory field, there's a table for data entry:

パス名	座標	回転
Name	Position	Rotation
<input type="text"/>	x <input type="text"/>	x <input type="text"/>
	y <input type="text"/>	y <input type="text"/>
	z <input type="text"/>	z <input type="text"/>

Below the table is a large text area for the JSON output. To the right of the text area are three buttons: "リストに追加" (Add to list), "選択項目を削除" (Delete selected items), and "データをすべてクリア" (Clear all data). At the bottom right of the window is a yellow icon.

クリックするとエクスプローラーが開かれる。

ファイル名を入力して保存する。
右図：書き出し例)

※データリストの行がすべて
以下のフォーマットで書き出される。
数値はdouble型。

```
{  
  Path : パス名,  
  Position : 座標,  
  Rotation : 回転  
}
```

```
[  
  {  
    "Path": "walk1",  
    "Position": {  
      "X": 0.0,  
      "Y": 0.0,  
      "Z": 10.0  
    },  
    "Rotation": {  
      "X": 0.0,  
      "Y": 0.0,  
      "Z": 0.0  
    }  
  },  
  {  
    "Path": "walk2",  
    "Position": {  
      "X": 0.0,  
      "Y": 0.0,  
      "Z": 50.0  
    },  
    "Rotation": {  
      "X": 0.0,  
      "Y": 0.0,  
      "Z": 0.0  
    }  
  }  
]
```

④パス名

主に行動名などを入力する。

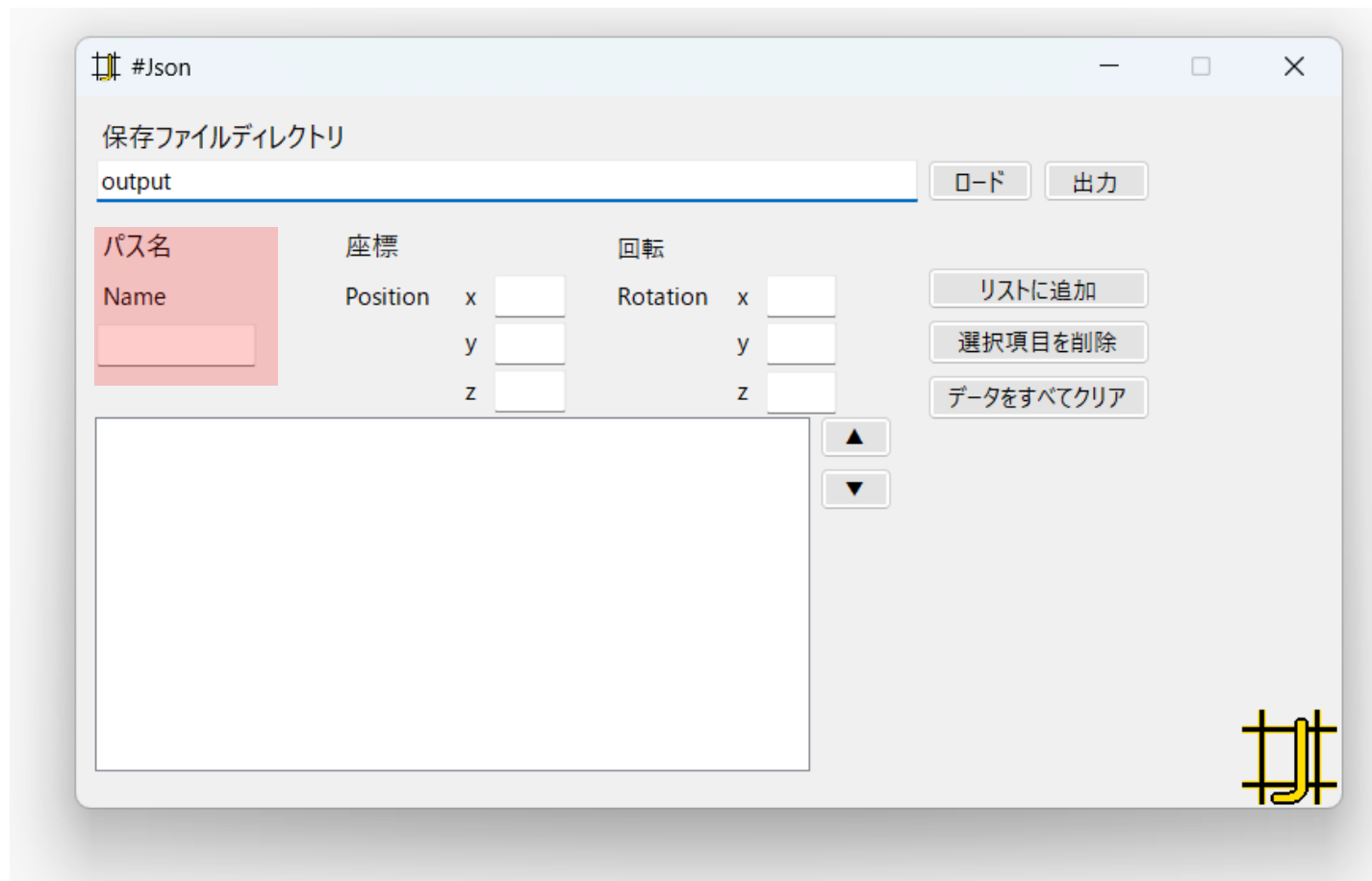
例)

walk1 … 歩行パス 1

walk2 … 歩行パス 2

【注意事項】

1. 日本語名非推奨



The screenshot shows the #Json application window. The title bar is "#Json". The main area has a "保存ファイルディレクトリ" (Save File Directory) field with "output" entered. Below this is a table with columns "パス名" (Path Name), "座標" (Coordinates), and "回転" (Rotation). The "パス名" column has a red highlight. The "座標" column has sub-columns "Position" and "x", "y", "z". The "回転" column has sub-columns "Rotation" and "x", "y", "z". There are input fields for each of these. To the right of the table are buttons: "ロード" (Load), "出力" (Output), "リストに追加" (Add to List), "選択項目を削除" (Delete Selected Item), and "データをすべてクリア" (Clear All Data). A large empty text area is at the bottom left of the main content area. A yellow icon is in the bottom right corner of the window.

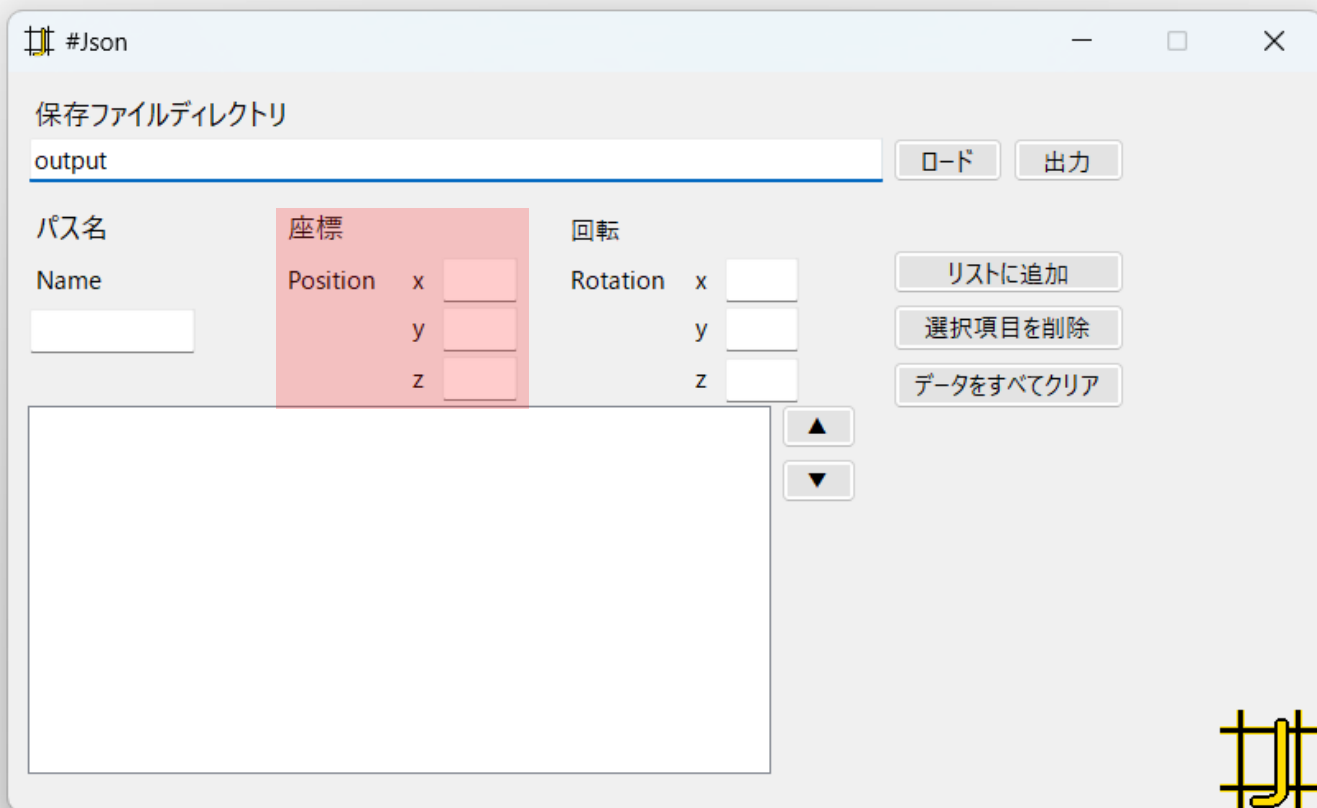
パス名	座標	回転
Name	Position	Rotation
	x	x
	y	y
	z	z

⑤座標パラメータ

そのパスに対応する座標を入力する。
何も入力されなかった場合、0自動で
入力される。

【注意事項】

1. double型で書き出される。
2. 文字列を入力すると、0が入力される。



The screenshot shows the #Json application window. At the top, there's a title bar with the icon and text "#Json". Below it, a "保存ファイルディレクトリ" (Save file directory) section contains a text field with "output" and two buttons: "ロード" (Load) and "出力" (Output). The main area is divided into two columns. The left column is titled "パス名" (Path name) and has a "Name" label and a text field. The right column is titled "座標" (Coordinate) and "回転" (Rotation). Under "座標", there's a "Position" label and three input fields for x, y, and z. Under "回転", there's a "Rotation" label and three input fields for x, y, and z. To the right of these input fields are three buttons: "リストに追加" (Add to list), "選択項目を削除" (Delete selected item), and "データをすべてクリア" (Clear all data). At the bottom right of the window, there's a yellow icon of a grid with a plus sign.

⑥回転パラメータ

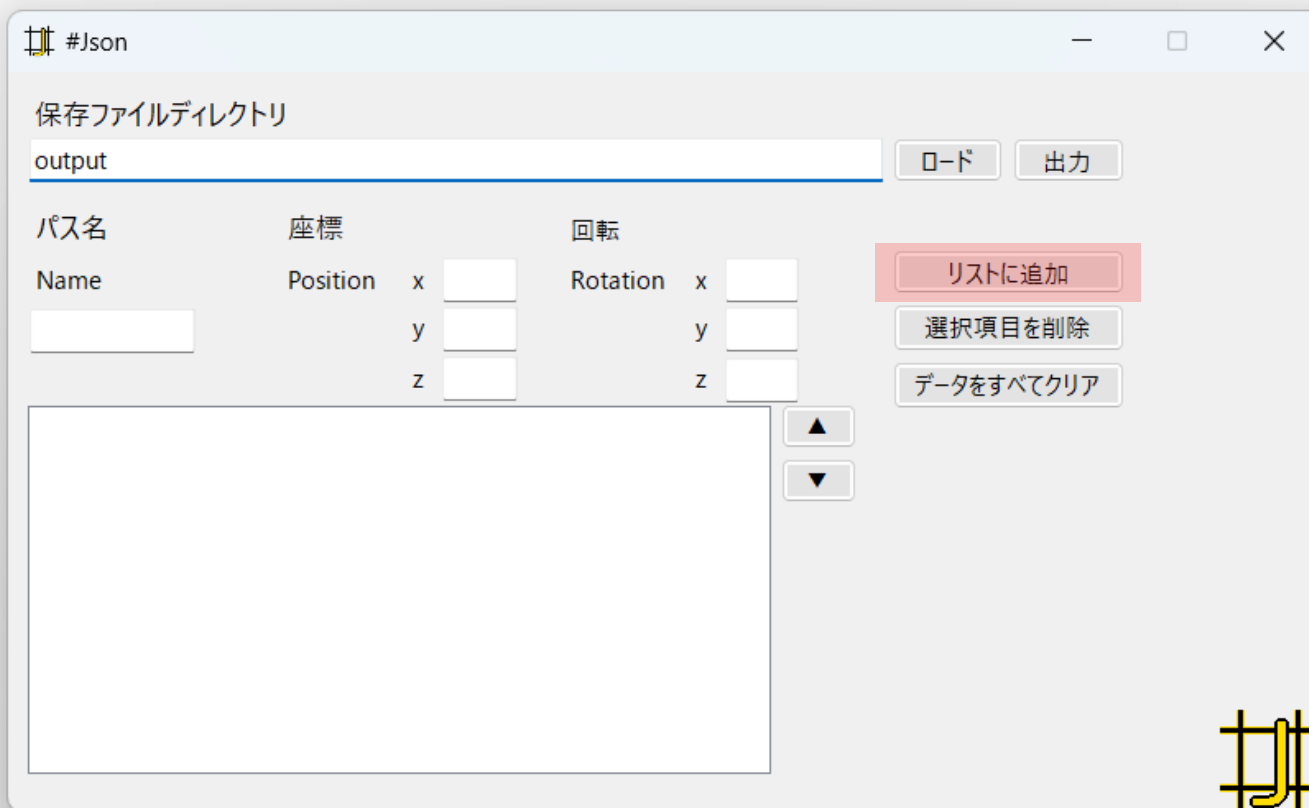
そのパスに対応する回転を入力する。
何も入力されなかった場合、0自動で
入力される。

【注意事項】

1. double型で書き出される。
2. 文字列を入力すると、0が入力される。

The screenshot shows the #Json application window. At the top, there's a title bar with the icon and text "#Json". Below it is a "保存ファイルディレクトリ" (Save file directory) section with a text input field containing "output" and two buttons: "ロード" (Load) and "出力" (Output). The main area is divided into two columns. The left column has a "パス名" (Path name) section with a "Name" label and an empty text input field. Below it is a large empty text area. The right column has a "座標" (Coordinate) section with labels "Position" and "x", "y", "z" and corresponding input fields. To the right of this is a red-shaded "回転" (Rotation) section with a label "Rotation" and input fields for "x", "y", and "z". To the right of the rotation section are three buttons: "リストに追加" (Add to list), "選択項目を削除" (Delete selected item), and "データをすべてクリア" (Clear all data). At the bottom right of the window is a yellow icon.

⑦リストに追加ボタン



The screenshot shows the #Json application window. At the top, there's a title bar with the icon and text '#Json'. Below it is a '保存ファイルディレクトリ' (Save file directory) section with a text input field containing 'output' and two buttons: 'ロード' (Load) and '出力' (Output). The main area is divided into two columns. The left column has a 'パス名' (Path name) section with a 'Name' label and an input field. The right column has a '座標' (Coordinate) section with 'Position' labels and three input fields for x, y, and z. Below these is a '回転' (Rotation) section with 'Rotation' labels and three input fields for x, y, and z. A large empty list box is at the bottom left. To the right of the list box are three buttons: 'リストに追加' (Add to list) which is highlighted with a red border, '選択項目を削除' (Delete selected item), and 'データをすべてクリア' (Clear all data). There are also up and down arrow buttons next to the list box.

入力されたデータを下のデータリストに追加する。

入力値がない場合は、以下の通りになる。

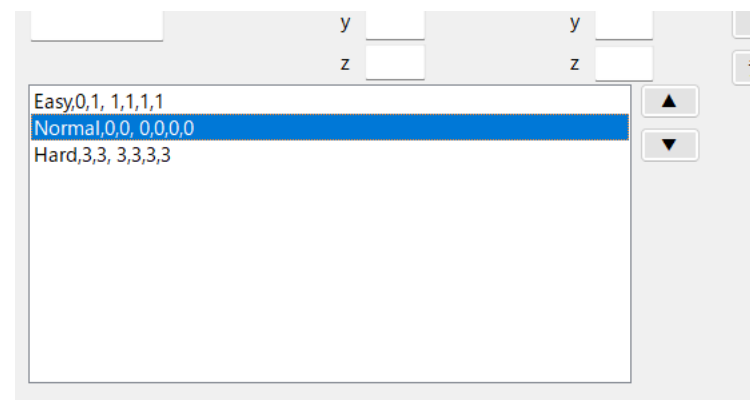
Name : noData

Position : 0.0,0.0,0.0

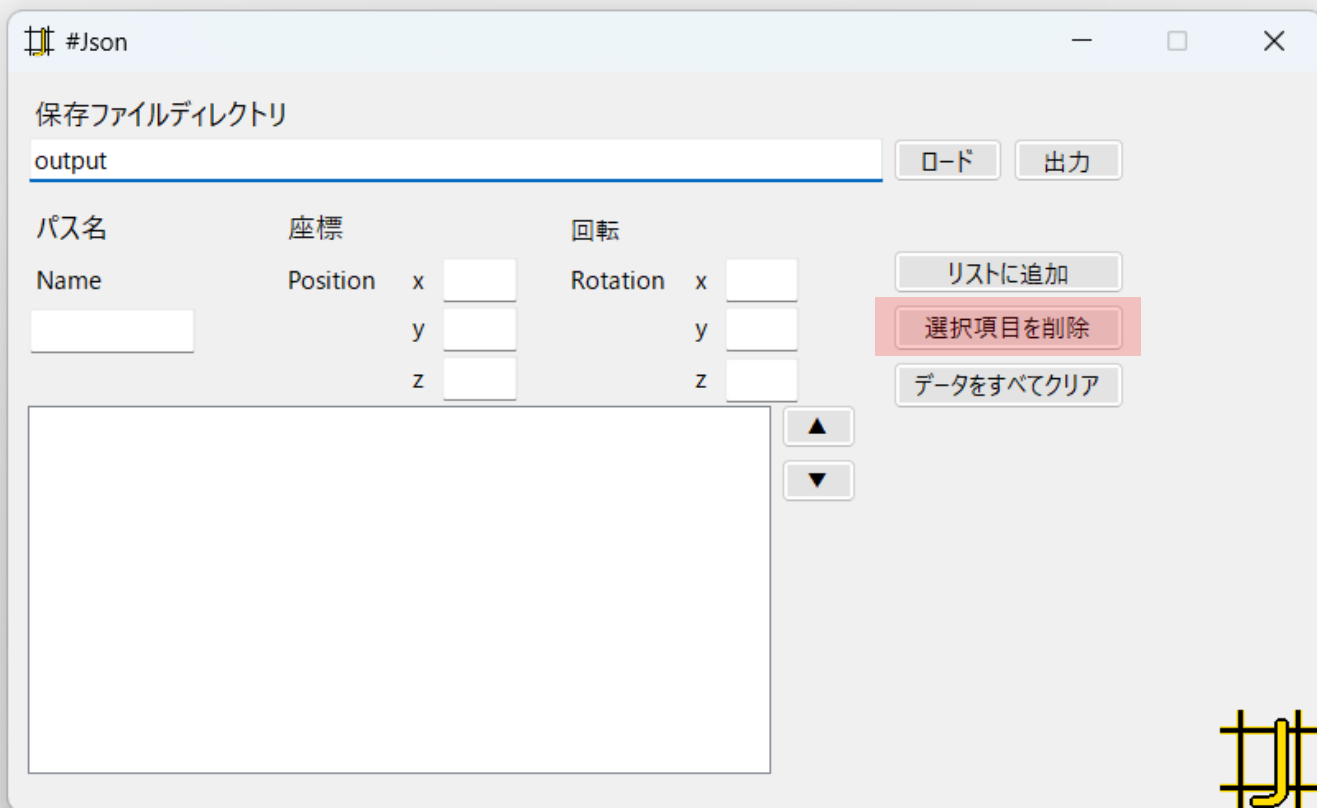
Rotation : 0.0,0.0,0.0

⑧選択項目を削除ボタン

データリストの選択している項目を削除する。



上記の場合だと、パス名「Normal」の一行が削除される。



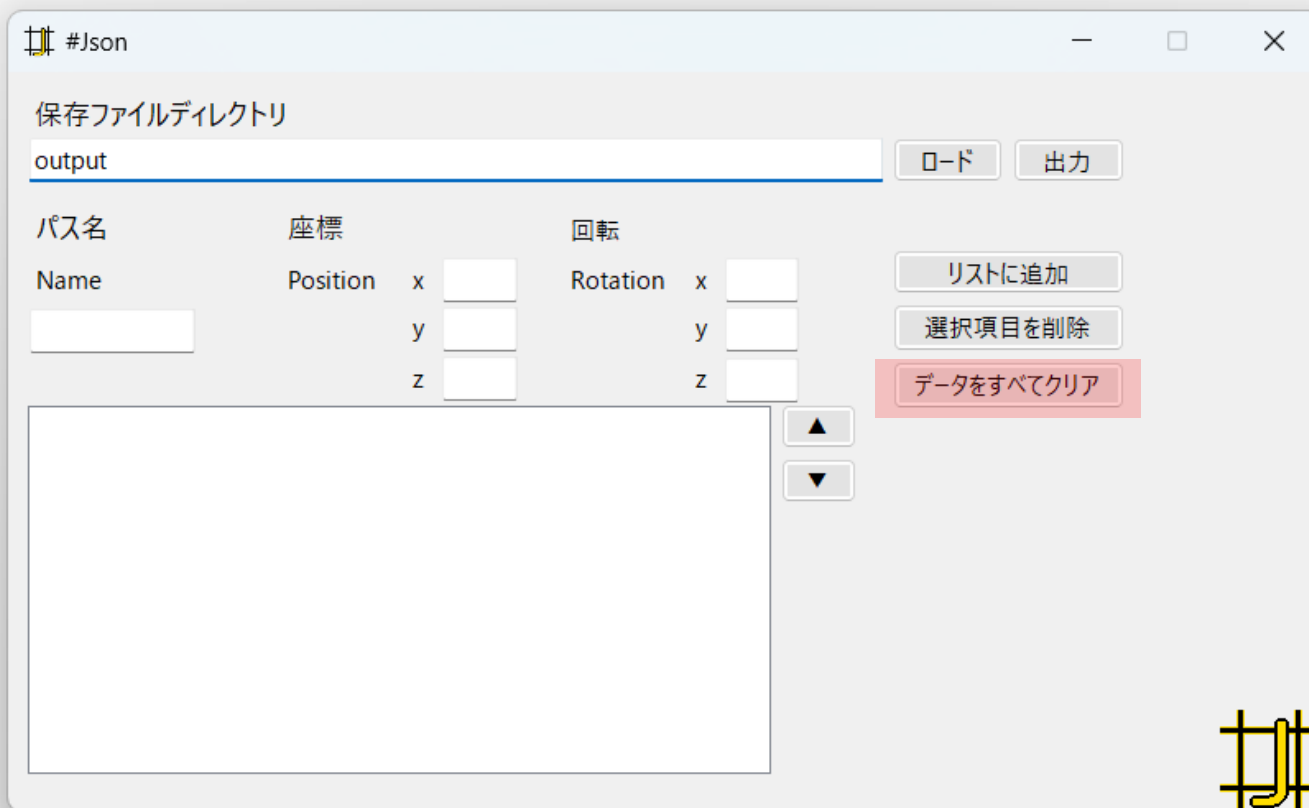
⑨データをすべてクリアボタン

データリスト、入力済みパラメータをすべて削除する。

起動時の初期状態に戻る。

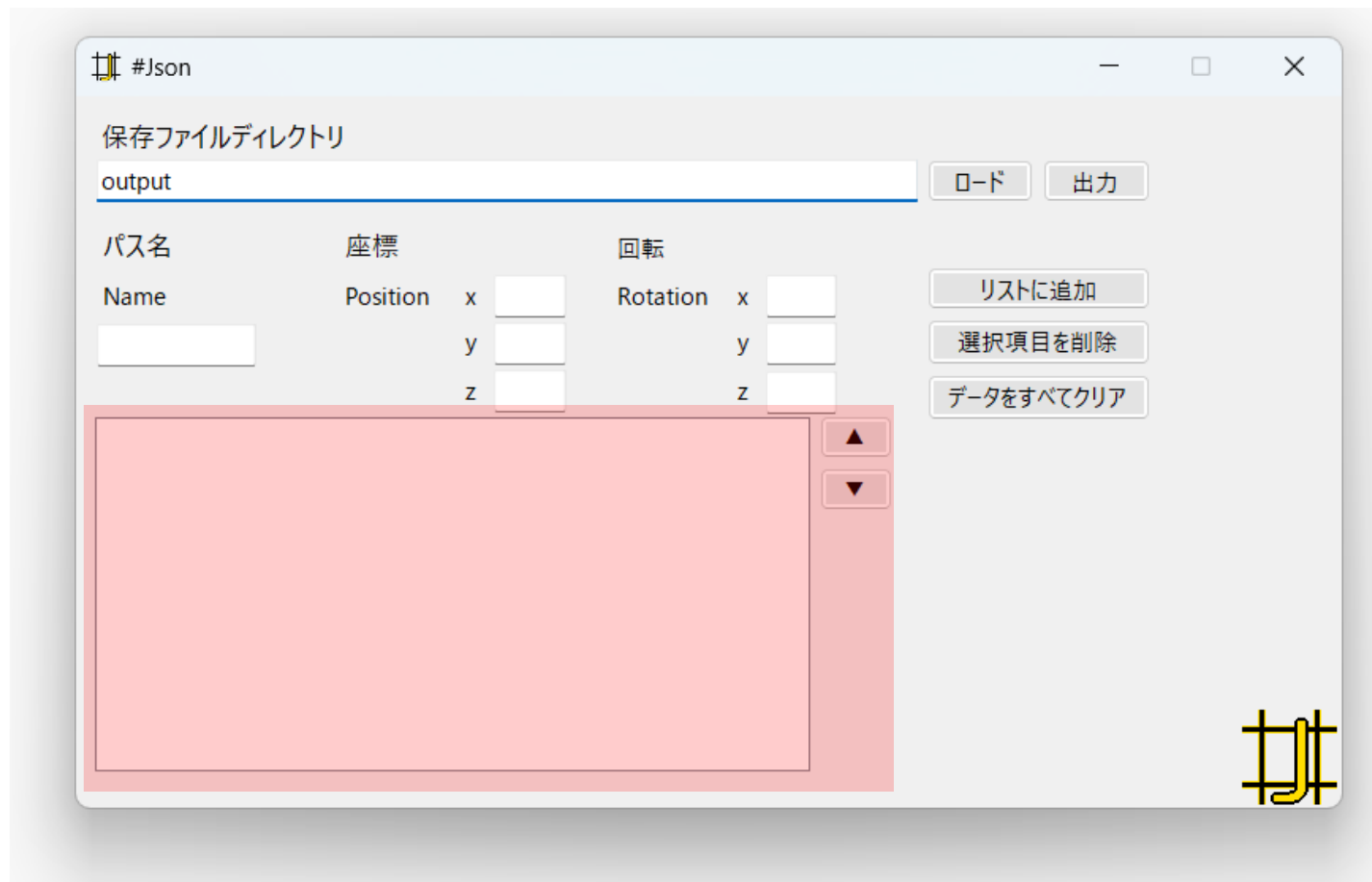
【注意事項】

1. 保存していないデータがあれば、保存しておく必要がある。



The screenshot shows the #Json application window. At the top, there's a title bar with the icon and text '#Json'. Below it, a '保存ファイルディレクトリ' (Save file directory) section contains a text field with 'output' and two buttons: 'ロード' (Load) and '出力' (Output). The main area is divided into two columns: 'パス名' (Path name) with a 'Name' label and an empty text field, and '座標' (Coordinates) with 'Position' labels and input fields for x, y, and z. To the right of these is a '回転' (Rotation) section with 'Rotation' labels and input fields for x, y, and z. Below the input fields are three buttons: 'リストに追加' (Add to list), '選択項目を削除' (Delete selected items), and 'データをすべてクリア' (Clear all data), which is highlighted with a red background. At the bottom left is a large empty rectangular area, and at the bottom right are two small arrow buttons (up and down) and a yellow icon.

⑩出力データリスト・順番移動ボタン

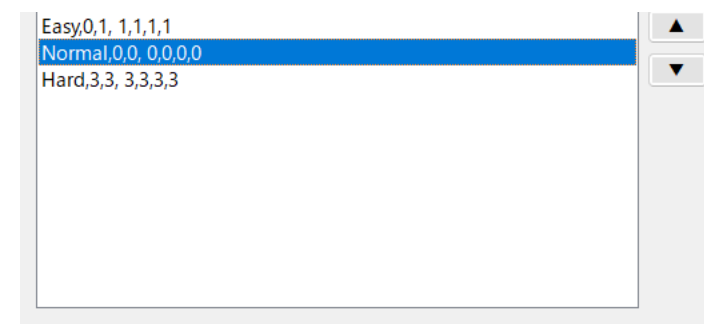


The screenshot shows the #Json application window. At the top, there's a title bar with the icon and text "#Json". Below it is a "保存ファイルディレクトリ" (Save file directory) section with a text field containing "output" and "ロード" (Load) and "出力" (Output) buttons. The main area is a table with columns for "パス名" (Path name), "座標" (Coordinates), and "回転" (Rotation). The table has three rows: "Name", "Position", and "Rotation", each with sub-columns for x, y, and z. Below the table is a large red rectangular area. To the right of the table are three buttons: "リストに追加" (Add to list), "選択項目を削除" (Delete selected item), and "データをすべてクリア" (Clear all data). At the bottom right of the window is a yellow icon.

パス名	座標	回転
Name	Position	Rotation
	x	x
	y	y
	z	z

出力時に書き出されるデータリスト。

上下ボタンで選択項目を対応する並び替える。



The screenshot shows a list of three items: "Easy,0,1, 1,1,1,1", "Normal,0,0, 0,0,0,0", and "Hard,3,3, 3,3,3,3". The "Normal" item is selected. To the right of the list are two buttons: an up arrow (▲) and a down arrow (▼).

上記の状態で▲ボタンを押すと「Easy」の上に移動し、▼ボタンを押すと「Hard」の下に移動する。

データ取扱例 (Json for Modern C++の場合)

```
//=====
// 読み込み
//=====
bool JsonHelper::Load(const wchar_t* path)
{
    // Jsonファイルの読み込み
    std::ifstream _ifs(path, std::ios::in);

    if (!_ifs) return EXIT_FAILURE;

    // Jsonデータの取り出し
    _ifs >> m_json;

    // Jsonファイルを閉じる
    _ifs.close();

    // パスを保存して正常終了
    m_path = path;
    return EXIT_SUCCESS;
}

//=====
// 書き出し
//=====
bool JsonHelper::Write(std::string str)
{
    // Jsonファイルの書き込み
    std::ofstream _ofs(m_path, std::ios::out);

    if (!_ofs) return EXIT_FAILURE;

    // Jsonデータの書き出し
    _ofs << str;

    // Jsonファイルを閉じる
    _ofs.close();

    return EXIT_SUCCESS;
}
```

```
JsonHelper json;
```

```
json.Load(L" ファイル名" );
```

```
Json data = json.GetData();
```

```
cout << data[0];
```

実行結果

Path : パス名, Position : 座標, Rotation : 回転