KI Sign

カテナスの
デジタルサイネージソリューション

チュートリアル編

KI Sign Editor で作る
表データ（リスト表示/行指定）



もくじ

表データ（リスト表示／行指定）について …… P.03

本チュートリアルの目的
対象
表データ（リスト表示／行指定）２種の機能について

サンプルのシナリオ／表データ（エクセル、CSV）の格納場所 …… P.04

サンプルのシナリオについて
対応するファイル形式／表データの格納場所を決める

シナリオの新規作成　表データ（リスト表示） …… P.05

1．エクセルファイルを作成する
2．シナリオを新規作成する
3．表データ（リスト表示）をドラッグ＆ドロップする
4．パスを設定する
5．表データのレイアウトを設定する
6．日付・時間型を変換する（抽出前処理）
7．セルの表示内容を設定する（セル設定）
8．日付・時間によって表示する条件を設定する（抽出条件１、抽出条件２）
9．データの並び替え（ソート）を設定する（抽出後処理）

補 足 …… P.15

１ページ当たりの表示行数を変更するには？
１ページ当たりの表示時間を変更するには？

表データを使ったシナリオテンプレート…… P.16

リハーサルをする／アップロード・配信をする…… P.17

付 録 …… P.18

関数一覧（文字列関数、数学関数、時間関数）

表データ（リスト表示／行指定）について

■ 本チュートリアルの目的

本チュートリアルは、お持ちのエクセル・CSV データを KI Sign Editor で表示処理するシナリオを新規に作成することによって、表データ（リスト表示 / 行指定）の機能を理解することを目的にしています。

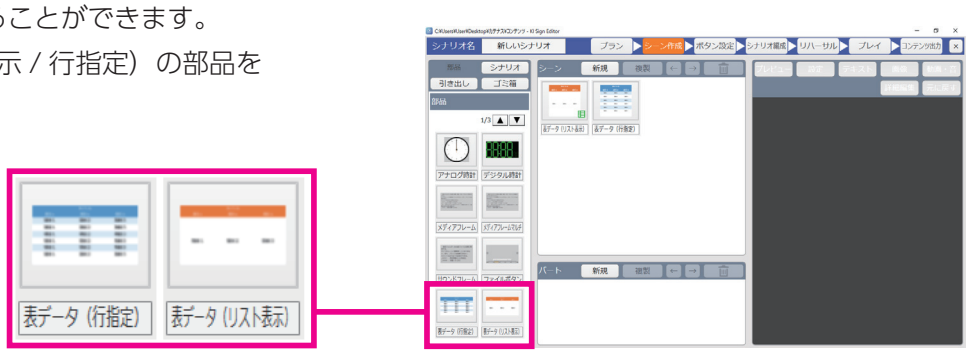
■対象

表データの機能内で KI 言語のスクリプトを使用しています。
プログラミングの初歩的な知識があるとよりわかりやすい構成になっています。
プログラミングが苦手な方もチュートリアル内に記載例の KI 言語のスクリプトをコピー＆ペーストすることで簡単にプログラムを設定をすることができます。
KI 言語 の詳細については「KI Language について」の HP を参照ください。
＜ https://ki.kathenas.com ＞

■表データ（リスト表示／行指定） 2 種の機能について

表データ（リスト表示 / 行指定）機能は、Excel ファイルまたは CSV ファイルから取得した値を元に、全ての表示行のセル内容を設定表示することができます。

KI Sign では標準で表データ（リスト表示 / 行指定）の部品を利用いただけます。



・表データ（リスト表示）：
　データを一覧形式で表示する場合に使用します。
　「行指定」と異なり全ての行に共通の設定となります。
　表示するデータ数が、1 ページに表示する行数を超えると、複数ページにわたって表示されます。
　それらのページは、設定した時間で順に切り替わります。

・表データ（行指定）：
　データの行を指定して表示する場合に使用します。
　「リスト表示」と異なり、表示行によって表示項目や表示スタイルを変更することができます。
　最大表示行数は固定となります。

表データ（リスト表示）イメージ

本日の催し物　05月01日（水）

時間	場所	催し物	利用者
09:00～12:00	中会議室	プロジェクトの進捗報告会	プロジェクトマネージメント株式会社
13:00～16:00	中会議室	チームビルディングのワークショップ	チームビルドソリューションズ株式会社
17:00～20:00	中会議室	新製品の発表会	イノベーションプロダクツ株式会社

場所	日付	開始時間	終了時間	催し物	利用者
中会議室	2024/5/1	09:00	12:00	プロジェクトの進捗報告会	プロジェクトマネージメント株式会社
中会議室	2024/5/1	13:00	16:00	チームビルディングのワークショップ	チームビルドソリューションズ株式会社
中会議室	2024/5/1	17:00	20:00	新製品の発表会	イノベーションプロダクツ株式会社

表データ（Excel・CSV）

表データ（行指定）イメージ



最終事故発生日	年度	重大災害	休業災害	不休災害
2023/1/2	2024	1	0	0

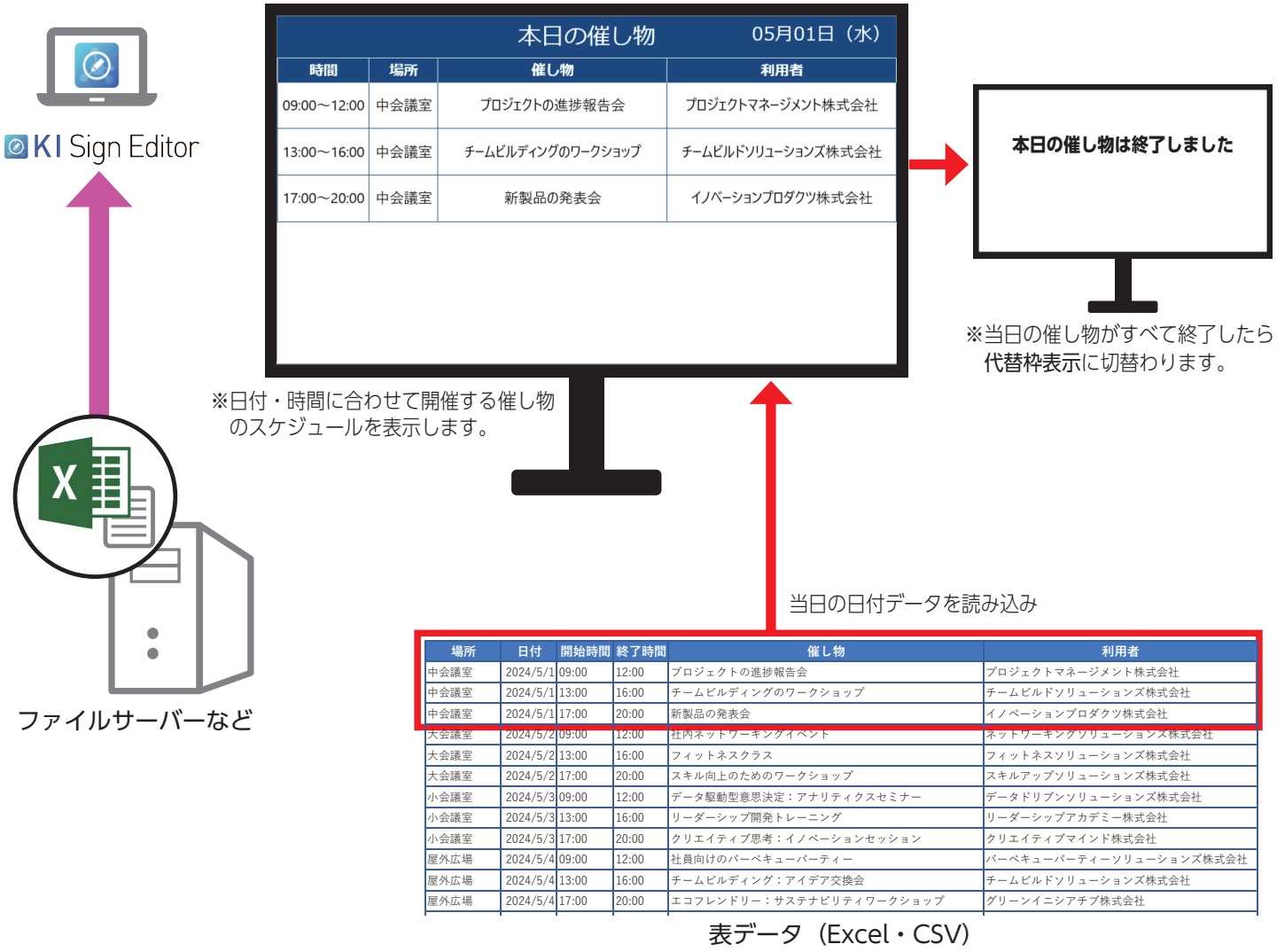
表データ（Excel・CSV）

本チュートリアルでは、表データ（リスト表示）機能を使った新規シナリオを作成する説明になっています。

サンプルのシナリオ/表データ(エクセル、CSV)の格納場所

■ サンプルのシナリオについて

例題サンプルとして作成するのは、表データ（リスト表示）機能を使って表データ（Excel・CSV ファイル）から取得した値を表示設定するシナリオです。時間・場所・催し物・利用者の情報を、日付・開始時間・終了時間に従って表示します。ファイルサーバーの共通フォルダなどに保存された表データから自動的に読み込み・更新するシステムを構築することができます。



■ 対応するファイル形式／表データの格納場所を決める

1．対応するファイル形式は下記の通りです。

・Excel ファイル　(拡張子が .xlsx の Excel 2007 以降のファイル形式)
・CSV ファイル　(文字コード Shift-JIS)

2．表データの格納場所を決めてください。

格納する場所(フォルダー)を決めて、表データを準備してください。
例)
C:¥Users¥ 演習 ¥sample.xlsx

※表データのパス設定は、**4．パスを設定する**で行います。

シナリオの新規作成　表データ（リスト表示）

表データ（リスト表示）機能を使って Excel・CSV データを表示するシナリオを新規に作成します。

1．エクセルファイルを作成する

シナリオに使用する表データ（エクセルファイル）を準備・作成します。
本チュートリアルではサンプルとして「sample.xlsx」を使用します。
表の構成は、場所・日付・開始時間・終了時間・催し物・利用者になっています。

場所	日付	開始時間	終了時間	催し物	利用者
中会議室	2024/5/1	09:00	12:00	プロジェクトの進捗報告会	プロジェクトマネージメント株式会社
中会議室	2024/5/1	13:00	16:00	チームビルディングのワークショップ	チームビルドソリューションズ株式会社
中会議室	2024/5/1	17:00	20:00	新製品の発表会	イノベーションプロダクツ株式会社

2．シナリオを新規作成する

［トップ画面］

操作手順

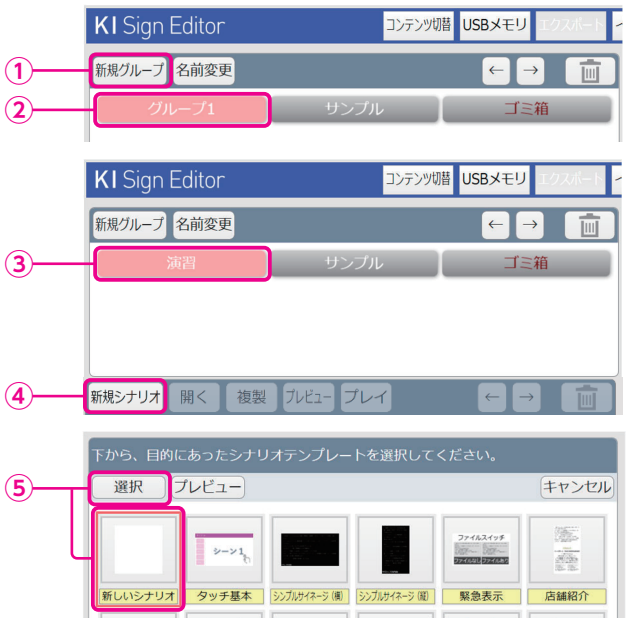
①**トップページ画面**左上の**新規グループ**をクリックする

②新規に作成したグループを選択し、**名前変更**をクリックする

③枠内にグループ名を入力し、エンターキーを押す
※グループ名が入力した名称に変更します。

④**新規シナリオ**をクリックする

⑤テンプレート一覧から**新しいシナリオ**を選択し、
選択ボタンをクリックする
※**プラン画面**が開きます。

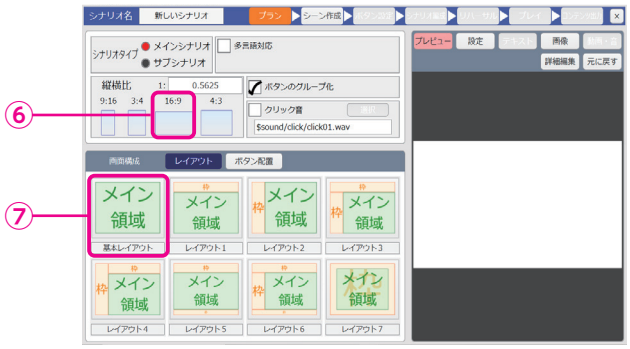


［プラン画面］

操作手順

⑥縦横比から**16:9**をクリックする

⑦レイアウト一覧から**基本レイアウト**をクリックする



3．表データ（リスト表示）をドラッグ＆ドロップする

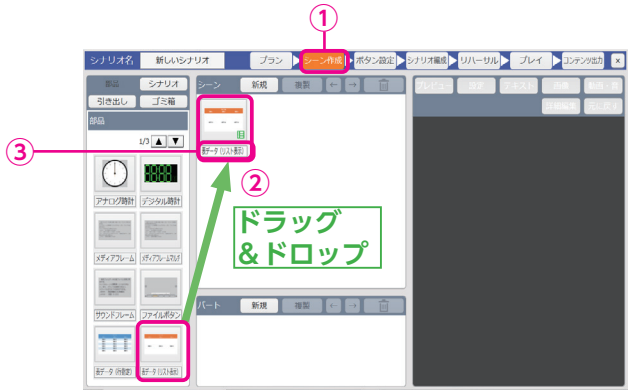
［シーン作成画面］

操作手順

①**シーン作成**をクリックする

②部品一覧から**表データ（リスト表示）**または
表データ（行指定）をシーン一覧へドラッグ＆ドロップする

③作成されたシーンのタイトル枠に名前を入力する
※例では「表データ（リスト表示）」のままです。



4．パスを設定する

［シーン作成画面］

操作手順

①**表データ（リスト表示）**をクリックする

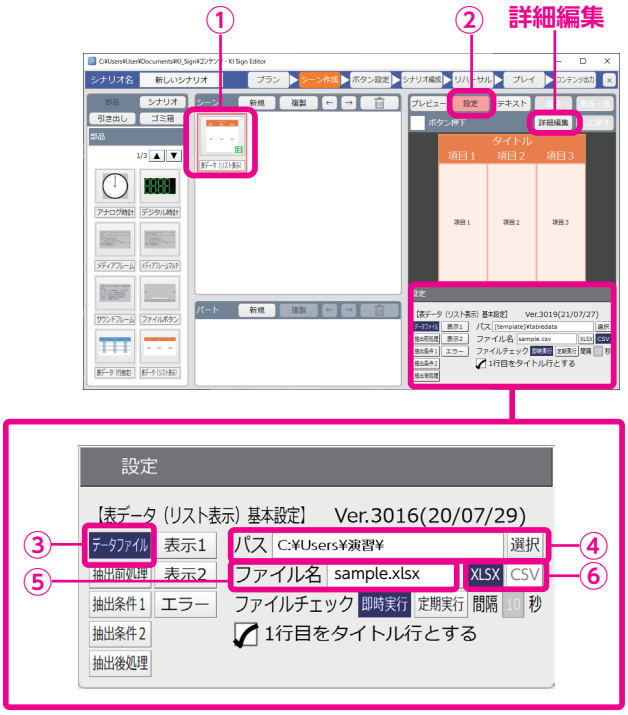
②**設定**ボタンをクリックする
※設定画面が表示されます。

③**データファイル**をクリックする
※データファイルの基本設定画面が表示します。

④**選択**をクリックし、表データの格納場所のパスを指定する

⑤ファイル名を指定する
⑥で選択されているファイルタイプに応じて拡張子が自動的に付与されます。

⑥ファイルタイプを指定します。



※シーン作成画面での設定画面の項目は、詳細編集画面内でも設定することができます。
詳細編集画面の枠一覧から**表データ（リスト表示）基本設定**を選択してください。

5．表データのレイアウトを設定する

表データ（Excel・CSV ファイル）から取得した値を表示する表のレイアウトを、詳細編集画面で行ってください。
セルの列数、テキスト内容、フォント、背景色、枠線、日付時間枠などを設定します。

5-1. セルの列数を設定する

［詳細編集画面］

操作手順

①**詳細編集**をクリックする
※詳細編集画面が開きます。

②枠一覧から**項目枠**の**枠**を選択する

③**複製**をクリックする
※枠一覧の枠が４列になります。

④枠一覧から**表データ（リスト表示）セル枠**を選択する

⑤**複製**をクリックする
※枠一覧のセル枠が４列になります。





項目枠、セル枠が４列になっていることを確認する



項目３を複製した場合

5-2. タイトル枠名、項目枠名を設定する

［詳細編集画面］

操作手順

①枠一覧から**タイトル枠**を選択する

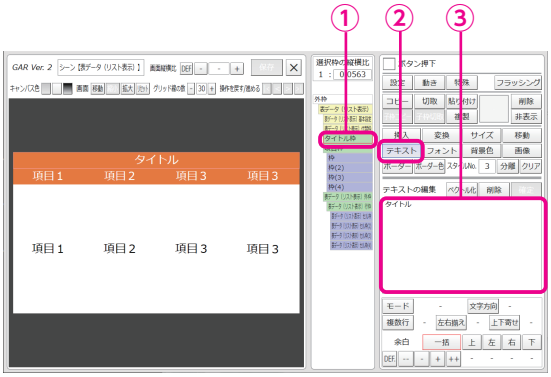
②**テキスト**をクリックする

③テキストの編集欄に”**本日の催し物**”と入力し、
確定をクリックする

④枠一覧から**項目枠**の**枠、枠（2）、枠（3）、枠（4）**を選択する

⑤テキストの編集欄にそれぞれの枠の名称を入力し、
確定をクリックする
※例では下記に設定

枠	時間
枠(2)	場所
枠(3)	催し物
枠(4)	利用者





5-3. タイトル枠、項目枠の背景色を設定する

［詳細編集画面］

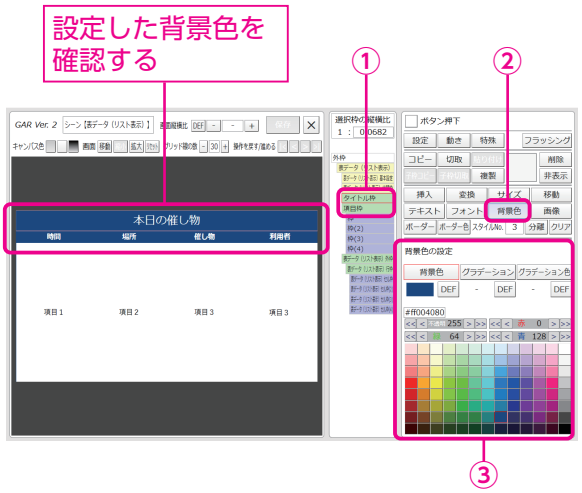
操作手順

①枠一覧から**タイトル枠**または**項目枠**を選択する

②**背景色**をクリックする
※背景色の設定画面が表示します。

③色テーブルまたは数値（不透明、赤、緑、青）で
設定する
※例では下記に設定

背景色	不透明 255、赤 0、緑 64、青 128
グラデーション	なし
グラデーション色	なし



5-4. 表の枠サイズ、フォントを設定する

表に表示される文字数から枠のサイズとフォントサイズを決定してください。

［詳細編集画面］



操作手順

①枠一覧からサイズを変更する枠を選択する

②**サイズ**をクリックする

③枠のサイズの **−−**、**−**、**+**、**++**ボタンをクリックする
※手入力で数値を入力するか、または調整パッド上でマウスをドラッグ＆ドロップすることによってもサイズを変更することができます。

④枠一覧からフォントを変更する枠を選択する

⑤**フォント**をクリックする

⑥文字色、文字種、シャドウ、その他（行間、文字サイズ、太さ）を設定する

※例での各サイズ・フォントは下記に設定

設定した枠サイズ、フォントを確認する



※編集ソフトウェアではプレイ時に画面のサイズがどのように変更されても、画面を構成する各パーツが領域からあふれて見えなくなることがないように、画面の変更に合わせてフォントサイズが自動的に調節されます。文字サイズは画面の大きさに対する相対的な大きさのことを指しています。

■サイズ

タイトル枠	1.2
項目枠	0.7
枠	0.4
枠(2)	0.3
枠(3)	1
枠(4)	1
表データ（リスト表示）外枠	8
表データ（リスト表示）行枠	1
表データ（リスト表示）セル枠	0.4
表データ（リスト表示）セル枠(2)	0.3
表データ（リスト表示）セル枠(3)	1
表データ（リスト表示）セル枠(4)	1

■フォント

［タイトル枠］

文字色	不透明 255、赤 255、緑 255、青 255
文字種	スタイル DEF、フォント DEF
シャドウ	なし
その他	文字サイズ 37、太さ DEF

［項目枠］ 枠、枠(2)、枠(3)、枠(4)

文字色	不透明 255、赤 255、緑 255、青 255
文字種	スタイル 太字、フォント DEF
シャドウ	なし
その他	文字サイズ 23、太さ DEF

［セル枠］ 表データ（リスト表示）セル枠

文字色	DEF
文字種	スタイル DEF、Yu Gothic UI
シャドウ	なし
その他	文字サイズ 23、太さ DEF

5-5. 枠線 (ボーダー) を設定する

［詳細編集画面］

操作手順

①枠一覧から枠線を設定する枠を選択する

②**ボーダー**をクリックする
※枠の形状とスタイル画面が表示します。

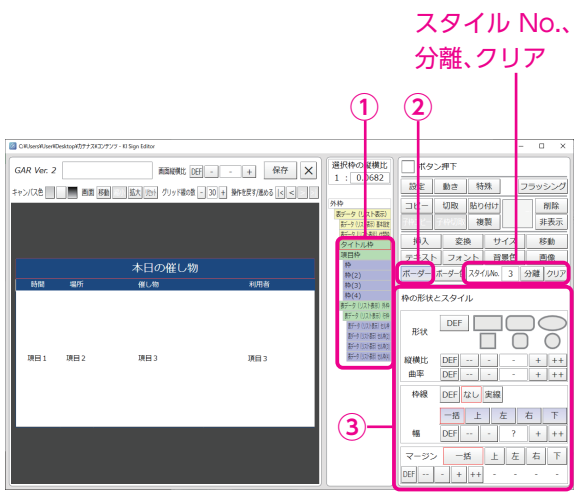
③**ボーダーの形状、枠線、幅、マージン**などを設定する

④枠一覧から枠線の色を設定する枠を選択する

⑤**ボーダー色**をクリックする

⑥枠線の色を設定する

※例での各ボーダー、ボーダー色は下記に設定





※ボーダー・ボーダー色の設定は、**スタイル No.** によって管理されています。
スタイル設定を修正した場合、同じ**スタイル No.** を持つ枠は、一斉に同じスタイル設定が適用されます。スタイルの設定を全て設定する必要がなく、スタイル No. を変えるだけで簡単に編集作業が行うことができます。

※枠を複製した場合、複製元の**スタイル No.** が複製先に引き継がれます。

※既存のスタイルから個別のスタイル設定に変えたい場合は、**分離、クリア**をクリックすることで新たな**スタイル No.** を作成できます。
・**分離**　：スタイル設定を引き継ぎ、新しい No. を作成します。
・**クリア**：スタイルが初期設定に戻り、新しい No. を作成します。

枠種	ボーダー［枠線］	ボーダー色
タイトル枠	無し	—
項目枠	実線　上 0、左 1、右 0、下 0	不透明 255、赤 255、緑 255、青 255
枠	実線　上 1、左 0、右 1、下 0	不透明 255、赤 255、緑 255、青 255
枠(2)	実線　上 1、左 0、右 1、下 0	不透明 255、赤 255、緑 255、青 255
枠(3)	実線　上 1、左 0、右 1、下 0	不透明 255、赤 255、緑 255、青 255
枠(4)	実線　上 1、左 0、右 1、下 0	不透明 255、赤 255、緑 255、青 255
表データ（リスト表示）外枠	無し	—
表データ（リスト表示）行枠	実線　上 0、左 1、右 0、下 0	不透明 255、赤 0、緑 64、青 128
表データ（リスト表示）セル枠	実線　上 0、左 0、右 1、下 1	不透明 255、赤 0、緑 64、青 128
表データ（リスト表示）セル枠(2)	実線　上 0、左 0、右 1、下 1	不透明 255、赤 0、緑 64、青 128
表データ（リスト表示）セル枠(3)	実線　上 0、左 0、右 1、下 1	不透明 255、赤 0、緑 64、青 128
表データ（リスト表示）セル枠(4)	実線　上 0、左 0、右 1、下 1	不透明 255、赤 0、緑 64、青 128

5-6. 日付時間枠を設定する

［詳細編集画面］

操作手順

①枠一覧から**タイトル枠**を選択する

②**挿入**をクリックする

③自由枠の挿入から**日付時間枠**を選択しクリックする
※枠一覧のタイトル枠の下に日付時間枠が追加されます。

④枠一覧から**日付時間枠**を選択する

⑤**設定**をクリックする

⑥日付・時間を設定する
※例では下記に設定
※日付・時刻の書式については**操作マニュアル**の83ページを参照してください。

表示内容	MM月dd日(ddd)
日付形式	西暦
元号定義	令和　2019/05/01

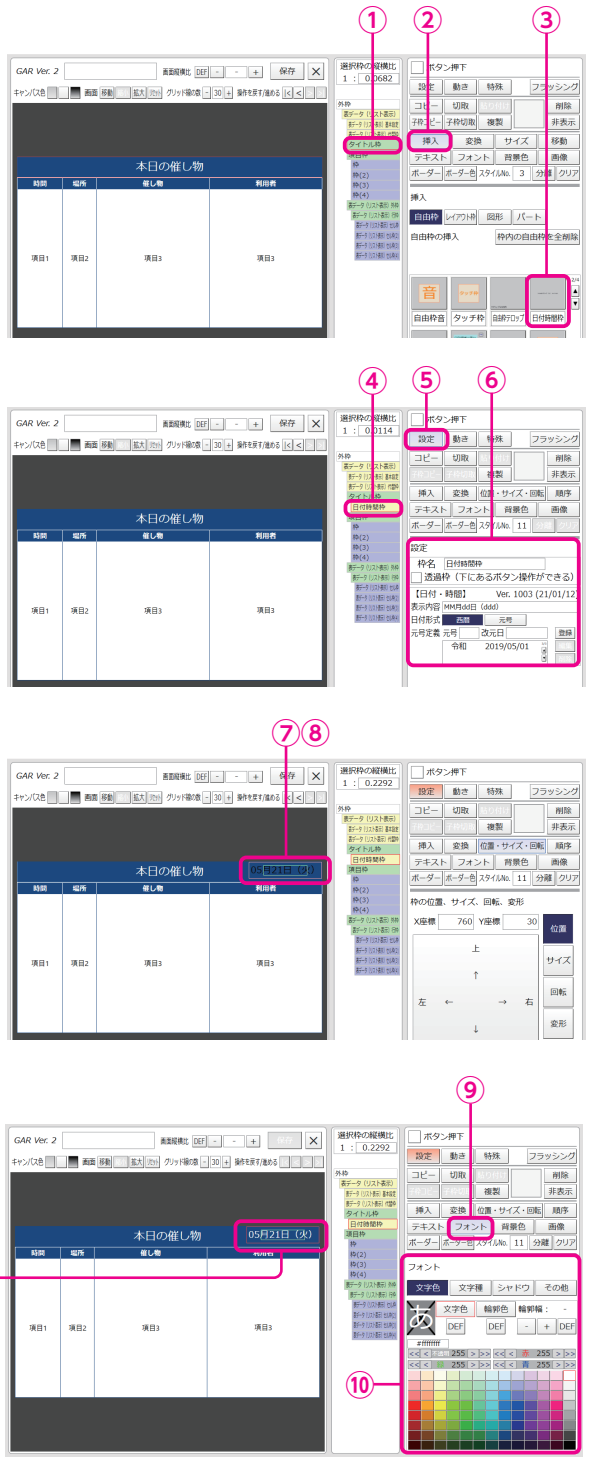
⑦日付時間枠をクリックする

⑧ドラッグ＆ドロップで位置・サイズを設定する

⑨**フォント**をクリックする

⑩文字色、文字種、シャドウ、その他（行間、文字サイズ、太さ）を設定する
※例では下記に設定

文字色	不透明 255、赤 255、緑 255、青 255
文字種	スタイル DEF、フォント DEF
シャドウ	なし
その他	文字サイズ 30、太さ DEF



5-7. 代替枠の表示内容を設定する

［詳細編集画面］

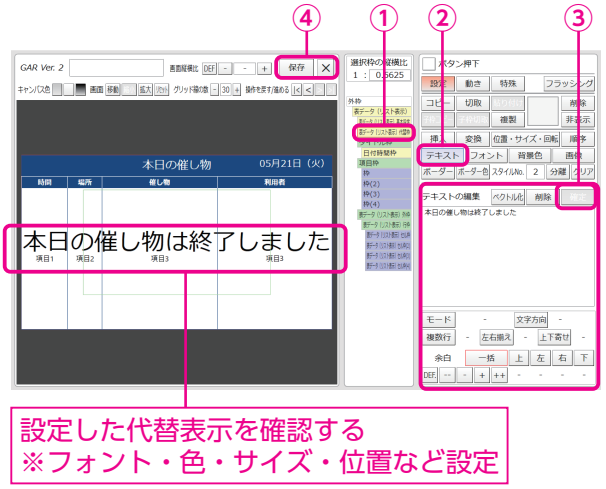
操作手順

①枠一覧から**表データ（リスト表示）代替枠**を選択する

②**テキスト**をクリックする

③テキストの編集欄に代替枠で表示するテキストを入力し、**確定**をクリックする
※代替枠で表示する背景、画像、位置など設定できます。
※例では「本日の催し物は終了しました」のテキストを入力しています。

④**保存**をクリックして設定を保存し、☒をクリックしてシーン作成画面に戻る



5-8. プレビューで表データのレイアウトを確認する

［シーン作成画面］［プレビュー］



操作手順

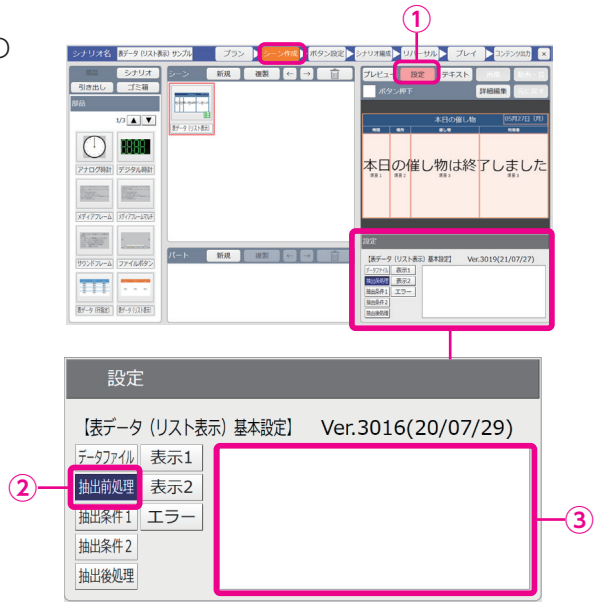
①**プレビュー**をクリックする
※詳細編集画面で設定した表データのレイアウトを確認する

※この時点では表データファイルは読み込まれていますが、まだ表示の割り当てが正しく行われていません。次項以降で設定を行います。

6．日付・時間型を変換する（抽出前処理）

抽出前に行う処理を記述します。抽出前処理には全角文字⇔半角文字の変換や日付・時間の書式変換等を記述します。

［シーン作成画面］［設定］



操作手順

①**設定**ボタンをクリックする

②**抽出前処理**をクリックする
※抽出前処理の設定画面が表示します。

③ KI 言語の記述を設定してください。
※例では下記③**抽出前処理**のスクリプトを設定しています。
表データは文字列として読み込まれますので、
日付型、時間型で扱いたい場合は、抽出前処理に
このスクリプトを必ず入れてください。
※③に下記の**スクリプト例**を入れ替え・組み合わせることで抽出前処理を変更することができます。
※関数一覧は**付録**を参照ください。

※③**抽出前処理**の"日付"、"開始時間"、"終了時間"はエクセルのタイトル名が入ります。これはタイトル名で表データの行を指定しています。**スクリプト例** rec[0]?は番号で表データの列を指定しています。
詳しくは、**7. セルの表示内容を設定する（セル設定）**の②**表示内容**を参照ください。（→12ページ）

設定	内容

③抽出前処理
 	
※日付型の場合は __ticks
時間型の場合は (time)__ticks
と記述してください。
	foreach((key)rec,table){
	try{
	rec.["日付"]?=__ticks(rec.["日付"]?);
	}catch((key)ex){
	logout%SysLog("Ticks変換に失敗したため、値を0に設定",rec.["日付"]?);
	rec.["日付"]?=0;
	}
	try{
	rec.["開始時間"]?=(time)__ticks(rec.["開始時間"]?);
	}catch((key)ex){
	logout%SysLog("Ticks変換に失敗したため、値を0に設定",rec.["開始時間"]?);
	rec.["開始時間"]?=0;
	}
	try{
	rec.["終了時間"]?=(time)__ticks(rec.["終了時間"]?);
	}catch((key)ex){
	logout%SysLog("Ticks変換に失敗したため、値を0に設定",rec.["終了時間"]?);
	rec.["終了時間"]?=0;
	}
	}

スクリプト例	例 1）1 列目の文字列を大文字に変換	foreach((key)rec,table){
		rec[0]?=toUpper(rec[0]?);
		}

	例 2）1 列目の日付データの書式を揃える	foreach((key)rec,table){
		try{
		rec[0]?=date(toTicks(rec[0]?));
		}
		}

7．セルの表示内容を設定する（セル設定）

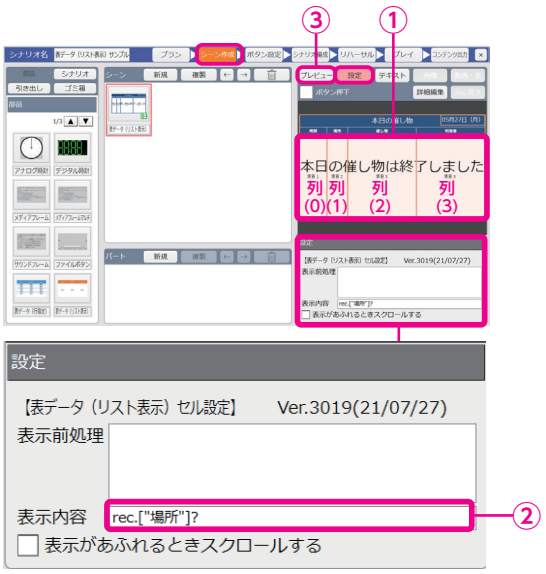
［シーン作成画面］［設定］［プレビュー］

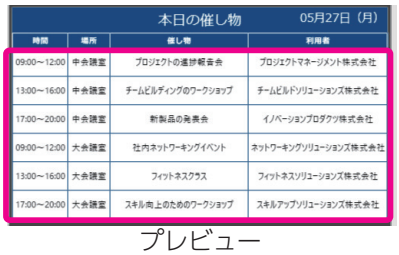
操作手順

①表示内容を設定する**セル**をクリックする
※列 [(0) ～ (3)] を選択する。
※表データ (リスト表示) セル設定画面が表示します。

②それぞれの列の表示内容を入力する

③**プレビュー**をクリックする
※セルの表示内容が表データの内容を表示しているか確認する
※関数一覧は**付録**を参照ください。





プレビュー

<table>
<tr><th>設定</th><th>内容</th></tr>
<tr><td>②表示内容</td><td>
表データからのセル値を取得・表示する内容を指定します。
取得方法は 2 つの方法（タイトル名で取得する、列番号で取得する）があります。
1）タイトル名で取得する方法
rec.["＜タイトル名＞"]? で指定します。
例：　rec.["場所"]?
※表データの 1 行目をタイトル行とする場合は、データファイルの基本設定
”1行目をタイトル行とする”がオンになっている必要があります。（初期設定：オン）
2）列番号で取得する方法
rec[＜列番号＞]? で指定します。（表データの列番号は 0, 1, 2, 3・・・で指定します。）
例（第 1 列の値 ”場所”）：　rec[0]?

※例では表データからタイトル名で取得し、下記に設定
<table>
<tr><td>列 (0)</td><td>format(rec.["開始時間"]?,":HH:mm")+"～"+format(rec.["終了時間"]?,":HH:mm")</td></tr>
<tr><td>列 (1)</td><td>rec.["場所"]?</td></tr>
<tr><td>列 (2)</td><td>rec.["催し物"]?</td></tr>
<tr><td>列 (3)</td><td>rec.["利用者"]?</td></tr>
</table>
※列(0)は[開始時間の列]＋[～]＋[終了時間の列]の表示になります。
例) 09:00～12:00

列 (1)　列 (0)　列 (2)　列 (3)
タイトル行→
<table>
<tr><th>場所</th><th>日付</th><th>開始時間</th><th>終了時間</th><th>催し物</th><th>利用者</th></tr>
<tr><td>中会議室</td><td>2024/5/1</td><td>09:00</td><td>12:00</td><td>プロジェクトの進捗報告会</td><td>プロジェクトマネージメント株式会社</td></tr>
<tr><td>中会議室</td><td>2024/5/1</td><td>13:00</td><td>16:00</td><td>チームビルディングのワークショップ</td><td>チームビルドソリューションズ株式会社</td></tr>
<tr><td>中会議室</td><td>2024/5/1</td><td>17:00</td><td>20:00</td><td>新製品の発表会</td><td>イノベーションプロダクツ株式会社</td></tr>
</table>
表データ「sample.xlsx」
</td></tr>
</table>

8．日付・時間によって表示する条件を設定する（抽出条件１、抽出条件２）

抽出条件を指定することにより、特定の日付・時間に一致する行のみを表示することが可能になります。

※**抽出条件１**と**抽出条件２**とはＡＮＤ条件です。

［シーン作成画面］［設定］



操作手順

①**抽出条件１**または**抽出条件２**をクリックする
※抽出条件 1, 2 の設定画面が表示します。
②に KI 言語の記述を設定してください。
※例では下記②**抽出条件**のスクリプトを入れています。
※②に下記の**スクリプト例**を入れ替えることで各種条件に変更することができます。
またそれらのスクリプト内の数値を修正することで、新たな条件設定に変更することができます。
※関数一覧は**付録**を参照ください。

設定	内容
②**抽出条件**	日付が一致し、現在の時間が終了時間以内の場合に条件を満たし表示します。 (date)rec.["日付"]?==(date)dateR()&&(time)timeR()<=(time)rec.["終了時間"]?

スクリプト例	例 1）現在日以降のデータのみ抽出 rec[0]?>=dateR() 1 行目をタイトル行としてデータを読み込んだ場合、項目名で条件を記述することもできます。 rec.["日付"]?>=dateR()
	例 2）当日のデータを取得 (date)rec.["日付"]?==(date)dateR()
	例 3）当日のデータかつ開始時間～終了時間のデータを取得 (date)rec.["日付"]?==(date)dateR()&&(time)rec.["開始時間"]?<=(time)timeR()&&(time)timeR()<=(time)rec.["終了時間"]?
	例 4）当日のデータかつ開始時間 30 分前～終了時間のデータを取得 (date)rec.["日付"]?==(date)dateR()&&(time)rec.["開始時間"]?-(time)"00:30"<=(time)timeR()&&(time)timeR()<=(time)rec.["終了時間"]?
	例 5）日付が直近 1 週間のデータを取得 (date)dateR()<=(date)rec.["日付"]?&&(date)rec.["日付"]?<=(date)dateR()+(time)"7.0:0"
	例 6）日付が直近 1 週間以内かつ当日の終了時間を経過していないデータを取得 (date)datetimeR()<=(date)rec.["日付"]?+(time)rec.["終了時間"]?&&(date)rec.["日程"]?<=(date)dateR()+(time)"7.0:0"
	例 7）日付が 1 カ月以内のデータを取得 (date)dateR()<=(date)rec.["日付"]?&&(date)rec.["日付"]?<=addMonths((date)dateR(),1)
	例 8）日付が 1 カ月以内かつ当日の終了時間を経過していないデータを取得 (date)datetimeR()<=(date)rec.["日付"]?+(time)rec.["終了時間"]?&&(date)rec.["日付"]?<=addMonths((date)dateR(),1)

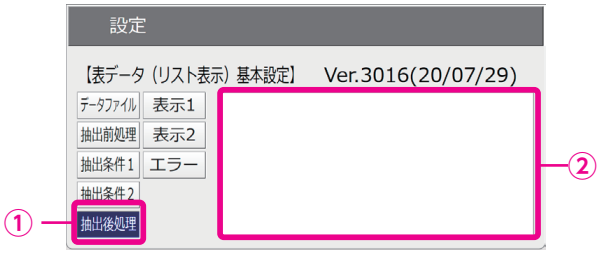
9．データの並び替え（ソート）を設定する（抽出後処理）

抽出後に行う処理を記載します。
データを並べ替えて表示する場合には、抽出後処理に記述します。

［シーン作成画面］［設定］

操作手順

①**抽出後処理**をクリックする
※抽出後処理の設定画面が表示します。
②に KI 言語の記述を設定してください。
※例は下記②**抽出後処理**のスクリプトを入れています。
※**スクリプト例**に開始時間の順番を並べ替える４パターンの例を示します。②にスクリプトを入れてください。
※関数一覧は**付録**を参照ください。



※抽出前処理で並び替えるデータの書式を揃えておかないと正しく並べ替えることができません。

設定	内容
②抽出後処理	二重ソート　開始時間で降順にソート 開始時間：昇順 (1)、催し物：降順 (0) ※二重ソートの場合、1 の位が最初のラベル、10 の位が 2 つ目のラベルに対応しますので、注意してください。 table%sort(".[¥"開始時間¥"],.[¥"催し物¥"]",01);
スクリプト例	例１）開始日の順番に並べ替える table%sort("[0]") または table%sort(".[¥"開始日¥"]")
	例２）開始時間で昇順にソート table%sort(".[¥"開始時間¥"]"); または table%sort(".[¥"開始時間¥"]",1);
	例３）開始時間で降順にソート table%sort(".[¥"開始時間¥"]",0);
	例４）二重ソート開始時間で降順にソート 開始時間：昇順 (1) 場所：降順 (0) table%sort(".[¥"開始時間¥"],.[¥"場所¥"]",01);

■1ページ当たりの表示行数を変更するには？

［シーン作成画面］［設定］

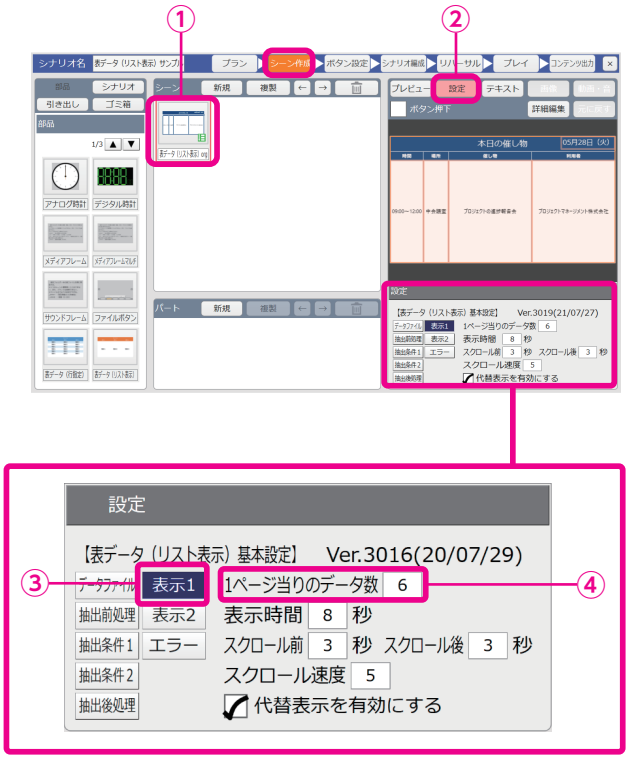
操作手順

①**表データ(リスト表示)**をクリックする

②**設定**ボタンをクリックする
※設定画面が表示されます。

③**表示 1** をクリックする
※表示 1 の設定画面が表示します。

④ **1 ページ当たりのデータ数**に数値を入力する



■1ページ当たりの表示時間を変更するには？

［シーン作成画面］［設定］

操作手順

①**表データ(リスト表示)**をクリックする

②**設定**ボタンをクリックする
※設定画面が表示されます。

③**表示 1** をクリックする
※表示 1 の設定画面が表示します。

④**表示時間**に数値を入力する



表データを使ったシナリオテンプレート

KI Editor で表データ（リスト表示、行指定）を組み込んだシナリオテンプレートは下記のとおりです。
これらのテンプレートをベースとして利用し、編集することで独自の表データ表示に対応することが可能です。
詳細は**操作マニュアル**を参照ください。

■操作マニュアル

シナリオテンプレート	表データ	ページ
(9)　イベント一覧	(リスト表示)	３４ページ
(10) イベント案内（行灯・横)	(行指定)	３５ページ
(11) イベント案内（行灯・縦)	(行指定)	３５ページ
(12) 宴席一覧	(リスト表示)	３６ページ
(13) 宴席案内（行灯・横)	(行指定)	３７ページ
(14) 宴席案内（行灯・縦)	(行指定)	３７ページ
(15) カレンダー	(リスト表示)	３８ページ
(16) リスト形式カレンダー	(リスト表示)	３９ページ

リハーサルをする／アップロード・配信をする

■ リハーサルを行い確認する

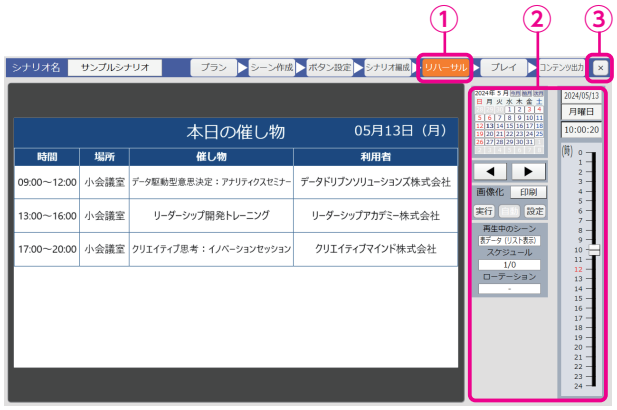
［リハーサル画面］

操作手順

①**リハーサル**をクリックする
※リハーサル画面が開きます。

②日付･時間を操作する
※設定された日付･時間にデータが正確に表示されるか確認してください。

②☒をクリックする
※シナリオを閉じます。



■ アップロード・配信をする

［トップページ画面］［LAN Manager 画面］

操作手順

①シナリオ一覧からシナリオを選択する
※例では「表データ（リスト表示）サンプル」を開きます。

②**アップロード**をクリックする

③**アップロード**をクリックする

④デスクトップにある

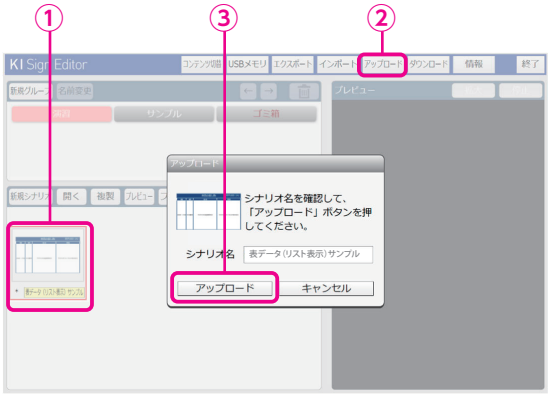
をダブルクリックして、LAN Managerを起動します。

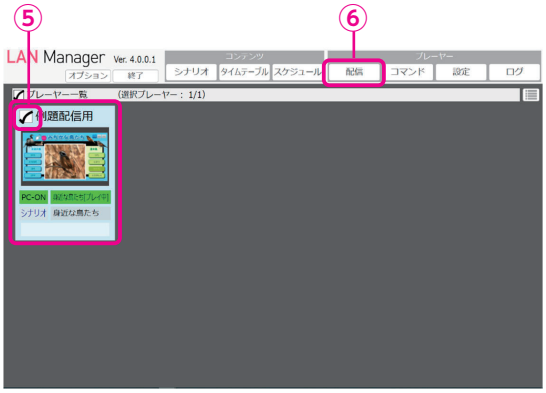
⑤配信を行うプレーヤーを選択する
（選択されると「✓」マークがつきます。）

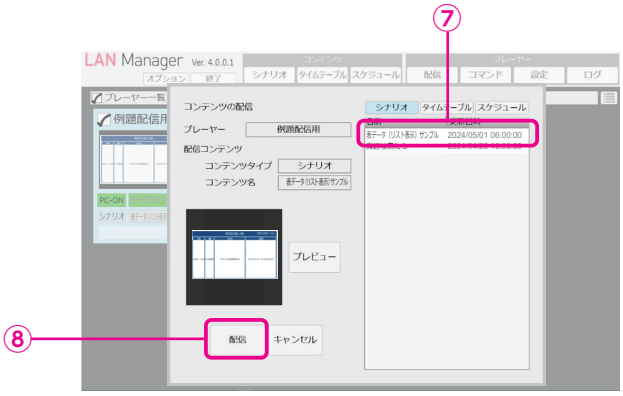
⑥**配信**をクリックする

⑦①でアップロードしたファイルを選択する
※例では「表データ（リスト表示）サンプル」です。

⑧**配信**をクリックする
※配信先に指定したプレーヤーに画像がローテーション表示されていることを確認します。







文字列関数一覧

関数名	説明	戻り値
charToCode((str)s) chrToCode((str)s)	文字 s をユニコードに変換する。	**(int)**
charType((str)s, [(int)idx]) chrType((str)s, [(int)idx])	文字列 s 中の idx 番目の文字のタイプを返す。 idx 省略時は、0 とみなされる。 返却時は以下の値が論理和された値が返却される。 (chrNarrow)0x1：半角文字 (chrControl)0x2：制御コード (chrAlphabet)0x4：アルファベット (chrUpper)0x8：大文字 (chrLower)0x10：小文字 (chrNumeral)0x20：数字 (chrSymbol)0x40：記号 (chrWide)0x80：全角 (chrHiragana)0x100：ひらがな (chrKatakana)0x200：カタカナ (chrKanji)0x400：漢字 (chrSpecial)0x8000：スペース (chrLetter)0x10000：文字	**(int)**
codeToChar((int)code) codeToChr((int)code)	ユニコード code を文字に変換する。	**(str)**
format((int or long or float)s, (str)fs)	fs の数値書式指定文字列に従い、s を文字列に変換する。 fs の先頭文字列が :(コロン) の場合には日時書式指定文字列として扱う。	**(str)**
format2((str)mult, (int or long or float)s...)	fs の複合書式指定文字列に従い s を文字列に変換する。	**(str)**
isInvalidFileName((str)s)	ファイル名として使用できるかを判定する。	**(int)**
isInvalidPathName((str)s)	パス名として使用できるかを判定する。	**(int)**
isNumeral((str)s)	文字列 s が数値かどうか判定する。	**(str)**
left((str)s, (int)count) strLeft((str)s, (int)count)	文字列 s の左から count 個からなる文字列を返す。	**(str)**
len((str)s) strLen((str)s)	文字列 s の長さを求める。	**(int)**
right((str)s, (int)count) strRight((str)s, (int)count)	文字列 s の末尾 count 個からなる文字列を返す。	**(str)**
strConv((str)s, (int)type)	文字列 s を type の値にしたがって様々に変換する。 type に指定できるのは、以下の値。 (chrNarrow)0x1：半角文字 (chrUpper)0x8：大文字 (chrLower)0x10：小文字 (chrWide)0x80：全角 (chrHiragana)0x100：ひらがな (chrKatakana)0x200：カタカナ	**(str)**
strFromBase64((str)s)	Base64 でエンコードされたデータをデコードする。	**(str)**
strIn((str)source, (str)search, [(int)from], [(int)count])	文字列 souce の先頭から from 以降に文字列 serch と一致する部分があれば、その最初の文字が先頭を 0 として数えたいくつ目であるかを返す。 count で、検査する文字の数を指定できる。	**(int)**
string((str)s, (int)count)	文字列 s が count 個繰り返す文字列を返す。	**(int)**
strInRev((str)source, (str)search, [(int)from], [(int)count])	strIn の文字検査を文字列 s の最後から行う。	**(int)**
strQuote((str)s)	文字列を引用符で囲む。	**(str)**
strReplace((str)s, (str)old, (str)new)	文字列 s 中から文字 old を探し、文字 new に置き換える。	**(str)**
strSplit((str)s, (str)sep)	文字列 s を文字 sep によって分割する。分割された部分文字列は、データキーツリーとして、返される。	**(key)**
strSub((str)s, (int)from, [(int)count])	文字列 s の前から from 番目から count 個からなる文字列を返す。 count 省略時、from 番目から最後の文字までを取り出す。	**(str)**
strToBase64((str)s)	文字列 s を Base64 でエンコードする。	**(str)**
strTrim((str)s)	文字列 s の前後にある空白文字を除去する。	**(str)**
strWidth((str)s)	文字列 s の幅を、半角 1、全角 2 として求める。	**(int)**
toLower((str)s)	文字列 s を大文字に変換する。	**(str)**
toUpper((str)s)	文字列 s を小文字に変換する。	**(str)**

数学関数一覧

関数名	説明	戻り値
abs((float)x)	絶対値を求める。	**(float)**
acos((float)x)	アークコサインを求める。	**(float)**
asin((float)x)	アークサインを求める。	**(float)**
atan((float)x)	アークタンジェントを求める。	**(float)**
ceiling((float)x)	切り捨てた値を求める。整数型、長整数型の場合は、 引数がそのまま返される。	**(float)**
cos((float)x)	コサインを求める。	**(float)**
exp((float)x)	e の x 乗を求める。	**(float)**
floor((float)x)	切り上げた値を求める。整数型、長整数型の場合は、 引数がそのまま返される。	**(float)**
ln((float)x)	自然対数を求める。	**(float)**
log((float)a, (float)b)	b を底とする、a の対数を求める。	**(float)**
max((int or long or float)a, (int or long or float)b, (int or long or float)c..)	引数で与えられた値の中で最大のものを求める。 引数の型は、最初の引数の型にキャストされる。 文字列型でも可。	**(var)**
min((int or long or float)a, (int or long or float)b, (int or long or float)c..)	引数で与えられた値の中で最小のものを求める。 引数の型は、最初の引数の型にキャストされる。 文字列型でも可。	**(var)**
mod((int or long or float)a, (int or long or float)b)	a を b で割ったときの余りを求める。	**(int or long or float)**
pow((float)a, (float)b)	a を b で累乗した値 (float) を求める。	**(float)**
rnd()	0 以上 1 未満の乱数を求める。	**(float)**
round((float)x)	四捨五入した値を求める。整数型、長整数型の場合は、 引数がそのまま返される。	**(float)**
sign((float)x)	符号を求める。	**(float)**
sin((float)x)	サインを求める。	**(float)**
sqrt((float)x)	平方根を求める。	**(float)**
tan((float)x)	タンジェントを求める。	**(float)**

時間関数一覧

関数名	説明	引数	戻り値
now()	現在の日付、時刻を時間数で返す。	なし	**(float)**
date()	時間数から日付を取り出し、yyyy/mm/dd の形式で返す。引数省略時は、現在の日付を返す。	**[(float)t]**	**(str)**
datetime()	時間数から日付を取り出し、yyyy/mm/dd HH:mm:ss の形式で返す。引数省略時は、現在の日付を返す。	**[(float)t]**	**(str)**
hour()	時間数から時間を取り出す。引数省略時は、現在の時刻から時間を返す。	**[(float)t]**	**(int)**
minute()	時間数から分を取り出す。引数省略時は、現在の時刻から分を返す。	**[(float)t]**	**(int)**
second()	時間数から秒を取り出す。引数省略時は、現在の時刻から秒を返す。	**[(float)t]**	**(int)**
year()	時間数から年を取り出す。引数省略時は、現在の日付から年を返す。	**[(float)t]**	**(int)**
month()	時間数から月を取り出す。引数省略時は、現在の日付から月を返す。	**[(float)t]**	**(int)**
day()	時間数から日を取り出す。引数省略時は、現在の日付から日を返す。	**[(float)t]**	**(int)**
time()	時間数から時刻を取り出し、hh:mm:ss の形式で返す。引数省略時は、現在の時刻を返す。	**[(float)t]**	**(str)**
toTicks()	文字列を時間数に変換する	**(str)**	**(float)t**
fromSerialDateTime()	時間数の起点を西暦 1 年 1 月 1 日 0 時 0 分 0 秒変換する。 主にエクセルファイルから取得した時間数を変換する際に使用します。	**(float)t**	**(float)t**
toSerialDateTime()	時間数の起点を西暦 1899 年 12 月 30 日 0 時 0 分 0 秒に変換する。 主にエクセルファイルから取得した時間数を変換する際に使用します。	**(float)t**	**(float)t**

KI 言語 の詳細については「KI Language について」の HP を参照ください。
< https://ki.kathenas.com >

Microsoft、Windows、Excel および Windows Media は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

S2406-0A