gusuku Customine カスタマインの歩き方

レコード更新編



はじめに(この冊子の目的)

使いこなすととても便利な gusuku Customine(以降カスタマインと表記)。 しかし、実現できることが多すぎるがゆえに使っていただいているお客様より 「難しい」というお声をいただくこともあります。そこで、実際に役立つカス タマイズを作成しながらカスタマインの使い方にも慣れていただくための教材 を用意しました。

この冊子では、様々なカスタマイズで利用可能な「レコード更新」機能に フォーカスして、「どのような要件のときに、どのやることを使用するべきか」 を理解していただくことを目的とします。様々な「レコード更新」方法を習得 していただくことで、実務に汎用的に活用していただければと思います。

この冊子で理解できること

レコード追加・レコード更新に関する「やること」の違い 「やること」ごとの注意点・制約 要件に応じた「やること」の選び方

目次

1.	はじめに理解していただきたいこと	Ρ	0	4
2.	レコード操作に関する基礎知識の整理・・・・・・・・・・・・・・・・・・・・・・・・	Ρ	1	7
3.	レコード追加、レコード更新に関する「やること」一挙解説	Ρ	2	2
4.	「やること」使い分けのポイント	Ρ	5	0
5.	さらにアプリを便利に活用するために	Ρ	5	2
6.	うまく動かないときは?	Ρ	5	4

1. はじめに理解していただきたいこと

1-1. 本書で使用する用語

本書ではカスタマインの画面や用語を以下のように記載します。

	ЭКУРАА	アクション		10 🔼 ···
	フィールドを無効化する	32.007	1両高・編集両面を表示した時	0
	7∢=∄⊁ (≙≥5		一覧商業の結果を押した時	一覧面面の影像を押した男は含め
	やること	条件	条件	
L .		DRITT	- 10. JULIN	

アクション

カスタマインの設定における 1 単位です。「やること」と「条件」を組み合わ せた 1 セットが 1 つのアクションとなります。

やること

カスタマイン設定画面の左側で選択する「やること」は、本書では[やること: フィールドを無効化する]のように記載します。

条件

カスタマイン設定画面の右側で選択する「条件」は、本書では<条件:一覧画 面を表示した時>のように記載します。

追加条件

「条件」の下部で「条件を追加」を選択して設定する「追加条件」は、本書で は<追加条件:フィールド値が特定の値ならば>のように記載します。

アクション番号

カスタマイン設定画面で各アクション(設定)の左上の数字です。本書内で【ア クション:1】と記載した場合には、アクション番号1番の設定を指します。

1 コメント入力		
アクション番号と無効化する		0
	フィールド 会社名	

パラメーター

設定する「やること」や「条件」に応じて設定する項目のことをパラメーター と記載します。本書内では《パラメーター:場所》のように記載します。

1 3xx+X#		11 2 1 11 100 A ····
ボタンをメニュ	2-位置に配置する 🥳	一覧画面を表示した時
	構成 - 気み高メニューの右側 ラベル レコードを取得する 通知性質 - 書右に追加する パラメーター	フィールド値が特定の値ならば マイールド 会社名 条件 等しい 比較値 (入力されていません) パラメーター 条件を追加

1-2. カスタマインの基本的な構造

カスタマインの基本的な構造は「~の時に~ならば~する」を [やること] と <条件 > を組み合わせて



という形で設定します。

この<条件>のドキュメント (<u>https://docs-customine.gusuku.io/ja/conditions/</u>)には「(追加条件)」と記載されている一覧があります。これは「~ならば」に該当するもので、<条件を追加>の箇所で指定します。 <条件>と<条件を追加>は指定する場所が違うのでご注意ください。

1-3. カスタマインで作成したカスタマイズが動く仕組み

カスタマインで作成した画面のカスタマイズは、ユーザーの画面操作に基づい て動く仕組みとなっています。

1-3-1. 画面ごとの特性

kintone にはいくつかの種類の画面があり、その画面ごとにできることが異なります。この画面ごとの特性を理解することで、よりスムーズにカスタマイズを作成できるようになります。

ー覧画面は一画面で同時に複数のレコードを閲覧できる、表示に特化した画面 です。そのため、カスタマイズにおいてもテーブルデータを表示できる一覧画 面の作成など、画面表示に関するカスタマイズに適しています。鉛筆ボタンを 押すことで編集もできますが、編集画面・追加画面と比較すると多くの制約が あります。編集する際には1レコード単位の編集しかできず、編集後は < 条件: ー覧画面を表示した時> に該当しないのでカスタマイズが動作しません。また、 一覧の編集画面以外の方法で値を変更した場合はリロードしない限り表示は変 わりません。

追加・編集画面ではフィールド値を変更することが出来ますが、保存ボタンを 押すまでは変更が確定していないのでキャンセルボタンを押すと変更内容は破 棄されます。

詳細画面は kintone の仕様ではフィールド値の更新は出来ないのですが、カ スタマインが裏側でレコード更新処理をする事でフィールド値を更新していま す。この場合は一覧画面での更新時と同じく値は変わっていますが表示は変わ らないのでリロードが必要になります。

1-3-2. 同じやることでも画面によって動作が違う

例として、[やること:フィールドに値をセットする] というやること1つを とっても、以下のように画面によって違いがあります。

追加画面の場合

フィールドに値がセットされて、表示も変わる

編集画面の場合

フィールドに値がセットされて、表示も変わる

詳細画面の場合

フィールドに値はセットされているが、表示は変わらない 表示を更新するためにはリロードする必要がある

一覧画面の場合

画面に表示されている全レコードのフィールドに値がセットされて、表示は 変わらない

表示を更新するためにはリロードする必要がある

一覧画面の鉛筆を押した編集画面

編集状態になっているレコードのフィールドに値がセットされて、表示も変わる

フィールド値がリアルタイムに変わるのは、kintone での操作と同じく追加画 面・編集画面のみということにご注意ください。

コラム:フィールド名とフィールドコード

kintone の仕組みとして、フィールド値を参照する場合は「フィールド コード」を指定します。

kintone のフィールドには「フィールド名」と「フィールドコード」が あるのでご注意ください。

簡単な見分け方としては、フィールドの個別設定画面で上にあるのが 「フィールド名」で下にあるのが「フィールドコード」です。

「フィールドコード」を明示的に指定しないと「文字列_1 行_1」のように自動でフィールドコードが設定されるため、フィールドコードがどのフィールドを指すのか判別しづらくなります。そのため、フィールド 名と同じ名称をつけたり、「フィールド名_アプリ名」のようにフィー ルドコードからフィールド名を類推しやすい名前にしておくことをおす すめします。

マールド名・ 文字列(行) コールド名を表示しない 自動計算する 自動計算する 必須項目にする 値の重複を禁止する 文字数(整数で指定) 最小 一 初期値 フィールドコード・ 文字列_1行_	☞ 文字列(1行)の	設定	×	
フィールド名* 文字列(1行) コィールド名を表示しない 自助計算する ② 必須項目にする 値の重複を禁止する 文字数(整数で指定) 最小 夏大 初期値 フィールドコード・ 文字列1行_ご	? ヘルプ			
文字列(1行) コールド名を表示しない 自動計算する (*) 必須項目にする 彼須項目にする 値の重複を禁止する 大字数(整数で指定) 最小 夏大 初期値 フィールドコード・ 文字列_1行_(*) キャンセル	フィールド名*			
 フィールド名を表示しない 自動計算する ② 必須項目にする 値の重複を禁止する 文字数(整数で指定) 最小 最大 初期値 フィールドコード・ 文字列_1行_ ご キャンセル 保存 	文字列 (1行)			
 自動計算する @ 必須項目にする 値の重複を禁止する 文字数 (整数で指定) 最小 最大 初期値 フィールドコード・ 文字列_1行_ご キャンセル 保存 	フィールド名を表示しない			
 	自動計算する 😰			
 値の重複を禁止する 文字数 (整数で指定) 最小 最大 初期値 フィールドコード・ 文字列_1行_ご 	必須項目にする			
文字数 (整数で指定) 最小 最大 初期値 フィールドコード* 文字列_1行_ご キャンセル 保存	値の重複を禁止する			
最小 最大 初期値 フィールドコード* 文字列_1行_ご キャンセル 保存	文字数 (整数で指定)			
初期値 フィールドコード* 文字列_1行_ ご キャンセル 保存	最小最大			
フィールドコード* 文字列_1行_ 📝 キャンセル 保存	初期值			
フィールドコード* 文字列_1行_ 🗹 キャンセル 保存				
キャンセル 保存	フィールドコード* 文字列_1行_ 📝			
	キャンセル		保存	

1-3-3. 自分が操作しているアプリでカスタマイズは動く

カスタマイズを作る際には「どのアプリで操作をしたときにカスタマイズを動 かしたいか」を意識して作成する必要があります。

先程の「フィールドに値をセット」だと、表示しているアプリのフィールドに 対してセットされますが、例えばこれを仮に別アプリのフィールドにセットし たい場合は「やること」が変わります。(「レコードを更新する(キーの値をフィー ルドで指定)」など)

コラム:これから「保存」するレコードのみが対象

カスタマインで作成したカスタマイズは、今後操作するときに適用され るものであるため、過去に登録したレコードには適用されません。

例えばあるアプリを作成ししばらく運用した後に、レコード保存後に ルックアップを自動更新するカスタマイズを作成したとしても、レコー ドを1つずつ保存していかないとルックアップは更新されません。です ので、カスタマイズ作成前に作成済みのレコードに対して更新しようと すると、大変な手間がかかり実用的ではありません。このような場合に は別の仕組みで更新する必要があります。

ルックアップの場合は、ルックアップフィールドの値を更新すれば取得 できるので、全レコードを取得してルックアップフィールドだけを書き 出せば可能です。また、CSV 読み込みで一括更新する方法でも可能です。

同じような話で、検索用文字列作成があります。これもレコード保存時 に文字列を作成しますが、過去のレコードは再保存が必要です。そして、 検索文字列の場合は CSV 読み込みでは作成できないので、何らかのカ スタマイズが必要になります。

本冊子では詳細な説明を記載しませんが、既に存在しているデータに対 して検索用文字列の生成を行いたい場合には、こちらのページをご参照 ください。 大量のレコードを安全に処理する方法:検索文字列を作成する事例 (紙の場合:サポートサイトで「検索文字列を作成する事例」で検索し てください) <u>https://support.gusuku.io/ja-JP/support/solutions/</u> <u>articles/36000268083</u>

また、kintone にはユーザーの画面操作を通知する「イベント」という仕組み があります。カスタマインで作成したカスタマイズは基本的に、この「イベン ト」に基づいて動きます。

1-3-4. イベント

イベントとは、「ユーザーの画面操作したタイミングでプログラムを動かすた めの kintone の仕組み」であり、一例として以下のようなイベントが定義さ れています

- 1. レコード追加画面を開いた時
- 2. プロセス管理のアクションを実行した時
- 3.フィールドの値が変わった時

kintone の「イベント」については cybozu developer network のヘルプを ご確認ください。

https://developer.cybozu.io/hc/ja/articles/360000361686

カスタマインにおける「条件」は主にこの「イベント」を日本語に置き換えた ものです。また、より使い勝手を向上するために kintone のイベント以外の ものも「条件」として利用できるようになっています。

カスタマイズを作成するときには「実現したいこと」とともに「どのイベント」 でその処理を実行したいかを意識し、「実現したいこと」→「やること」、「イ ベント」→「条件」に置き換えていただくとスムーズにカスタマイズが作れます。 例)レコード一覧画面表示後に注意すべき案件を色を変えて目立たせたい

実現したいこと:案件を色を変えて目立たせたい イベント:レコードー覧画面表示後 ↓ [やること:フィールド文字色を変更する] <条件:一覧画面を表示した時 >

例)入力時にユーザーに警告を促すために特定のフィールドの背景色を変えた い

実現したいこと:背景色を変えて警告を促したい イベント:フィールド値変更 ↓ [やること:フィールド背景色を変更する]

<条件:フィールドの値を編集して値が変わった時>

また、イベントごとの特性や制約を理解することで、kintone の制約上実現で きないことでも代替手段を検討することができます。

kintone の特性や制約によって実現できないことの例をあげますと、

1. kintone 側でイベントがない場合

2. イベントが発生しない場合

3. そのイベントでは実行できないカスタマイズの場合

などがあり、例えば「フォームブリッジでデータが登録されたことを契機に カスタマイズを動かしたい」という要件は、「2. イベントが発生しない場合」 に該当するためカスタマイズが動きません。

また、上記以外でも kintone 基本機能の「ファイルから読み込む」を使ってデー タを読み込んだ際にもカスタマイズが動きませんが、 これも kintone の制約 によるものです。

1-4. 実行順序

カスタマインでアクションを作成すると、番号が自動的に付与されます。 これはアクション番号と呼ばれるもので、カスタマイズの実行順とは関係あり ません。

1	コメント入力		
アクミ	<mark>ショシ番号</mark> ≧無効化する		?
		フィールド 会社名	

全てのアクションは何かの「イベント」を起点に処理が開始され、

「他のアクションの実行が完了した時」や「確認・入力ダイアログで「OK」を 押した時」などで繋がっている順に処理されます。

アクションの実行順をわかりやすく確認するにはアクショングラフを表示して いただくのが良く、カスタマイズ作成途中や作成後に表示してこまめに確認す ることをおすすめします。

なお、上手く動かないケースでよくあるのが、線で繋がっていないアクション の結果を参照したり、アクションの繋ぎ先を間違えている場合です。

特にアクションを複製した場合に設定を変え忘れているというケースは良くあ ります。 また、以下のように<条件:他のアクションの実行が完了した時>で同じアクション番号を指定したアクションが複数ある場合は、並行して処理されます。



また、これら複数のアクションが全て終わった時に次のアクションを続ける場合は、<条件:他のアクションの実行が完了した時 > で全てのアクションを指定してください。



その他にも <条件:いずれかのアクションの実行が完了した時 > を使うと、 指定したアクションのいずれかが完了した時にアクションが処理されます。 この条件は例えば、入力した金額によって処理を分ける場合などのようにどれ か1つが実行された後、同じ処理が続くような場合に使います。

案件接理 のカスタマイズ (3/2) ページ(2)	
332434	
なにもしない	レコード追加・編集画面を表示した時またはフィールドの値を編集して値が実
	わった時
	2x−4.F 0.8
ax>+3.8	
かにたしない	詳細高調を表示した時
9 - 9 6 6 s -	Statistic Chord Pro-
1957-530	
フィールドやグループを表示する	いずれかのアクションの実行が完了した時
フィールド アーブル	79932 6,7
	フィールド値が特定の値ならば
	7 e=8.6 10
	6# SL0
	12.859
コメント入力	いずれかのアクションの実行が完了した時
77-18 (-7).	794-14 67
21-01 1-22	
	フィールド値が特定の値でないならば
	フィールド 状態
	高作 参しい
	12.80 SM
ワクショングラフを閉じる マウスドラッグで移動、ホイールで拡大縮小ができます。	
レコード追加・編集画面を表示した時またはフィールドの値を	:編集して値が変わった時 詳細画面を表示した時
6	7
one	one one one

アクショングラフの詳しい使い方は「<u>6-4. 原因究明に役立つカスタマインの</u> <u>便利な機能</u>」に記載がありますので、併せてご確認ください。

8 4

* 9

2. レコード操作に関する基礎知識の整理

具体的な「やること」の解説に入る前に、カスタマインでレコードを扱う上で 必要となる用語を整理します。

2-1. レコード追加、レコード更新

レコード追加

kintone のアプリ上に新しいレコードを登録・作成することを「レコード追加」 と呼びます。本冊子の例では、「活動履歴」アプリで登録した内容をもとにし て「案件管理」アプリに転記したいと考えています。そのため、「レコード追加」 の機能を使用します。

安件 ID	安件夕	今菇	碎며
来住し	来ける	立识	1唯反
23-0001	粕玉商事SFA 導入案件	3,500,000円	С
23-0002	アールツー物産 パソコン入れ替え	980,000円	Α
23-0003	住井テクノロジー Web 改修	2,350,000円	В
23-0004	城不動産 複合機提案	530,000円	С
23-0005	賀寿丸建設 オフィス移転	8,500,000円	С
まだ存在しないレコードを新しく作る・・・追加			

レコード更新

kintone のアプリ上に既に存在するレコードを書き換える(上書きする)こと を「レコード更新」と呼びます。

上記の通り、本冊子の例では「活動履歴」アプリで登録した内容をもとにして 「案件管理」アプリに転記したいのですが、既に同一の案件が存在しているケー スがありえます。そのときには、新しいレコードを作成(レコード追加)する のではなく、既に登録されているレコードを書き換えたいので「レコード更新」 の機能を使用します。 「レコード更新」のときには、既に該当のレコードが登録されているか否か? を判断するための値が必要となります。それを「キー」と呼びます。 この例では、「案件 ID」フィールドが「キー」となります。そのため、「案件 ID」が既に登録されている場合には同一の案件とみなしてレコードを更新を行 います。

案件 ID	案件名	金額	確度
23-0001	粕玉商事SFA 導入案件	3,500,000円	С
23-0002	アールツー物産 パソコン入れ替え	980,000円	Α
23-0003	住井テクノロジー Web 改修	2,350,000円	В
23-0004	城不動産 複合機提案	530,000円	С
案件 ID が一致		BからA	に変更
23-0003	住井テクノロジー Web 改修	2,350,000円	А
既	に存在するレコードを書き換える	・・・更新	

2-2. フィールドマッピングの使い方

フィールドマッピングとは

複数のフィールドにまとめて値を入力するための機能を「フィールドマッピン グ」と呼びます。「フィールドマッピング」は本書で取り上げているレコード 追加・更新以外でも [やること:フィールド値をまとめてセットする] などの やることでも使用します。

フィールドマッピング画面の使用方法は<u>ドキュメント</u>にも記載がありますが、 以下に簡単な例とともに紹介します。

フィールドマッピング画面の見方

フィールドマッピング画面では、画面右側に指定されたフィールドに画面左側 で指定した値を入力します。また、画面上部の括弧内のアプリ名が入力先のア プリを指しています。

フィールドに入力したい値	このアク 入力した	ションの結果を <mark>こいフィールド</mark>
セットする値	セット先のフィ-	ールド
挿入 🖌	> 会社名	文字列(1行)
挿入 🖌	> 部署名	文字列(1行)
挿入 🖌	▶ 先方担当者	文字列(1行)
挿入 🗸	> TEL	文字列(1行)
挿入 🗸	> FAX	文字列(1行)
挿入 🗸	>メールアドレス	文字列(1行)
挿入 🗸	> 案件担当者名	ユーザー選択
挿入 🖌	▶ 見込み時期	日付
挿入 🖌	▶ 確度 *	ラジオボタン
挿入 🗸	▶製品名	ドロップダウン

入力できる値

フィールドマッピング画面の左辺では以下の値が指定可能です。

固定値

常に固定の値を指定したい場合には以下のように値を直接記述、もしくは = " 値 "の形式でしています。(どちらの記述方法も可能です)

以下の例では、常に製品コードに「C-0001」、製品名に「kintone」が入力されます。

セットする値 全て同じフ	ィールドコードをセット セット先の)フィールド
挿入 🗸 C-0001	▶ 製品コード	文字列(1行)
挿入 🖌 = "kintone"	> 製品名	文字列(1行)

フィールド

他のフィールドに入力されている値を指定したい場合には、以下のように「挿 入」のプルダウンよりフィールドを指定します。

セットする値	全て同じフィー
挿入 🖌	
フィールド	
他のアクションの結果	
関数	
挿入	
挿入 🖌	

以下の例は、「納品先会社名」というフィールドに対し、「請求書会社名」フィー ルドの値を入力することを示しています。

セットする値	全て同じフィールドコードをセット	セット先のフィ	ールド	
挿入 🖌 = 請求先会社	t名	請求先会社名	文字列(1行)	

他のアクションの結果

他のアクションの実行結果を使用することが可能です。例えば、「特定の日付 を取得する」の実行結果で得られる日付や、「入力ダイアログを表示する」によっ てユーザーに入力された値など様々な指定が可能です。

「他のアクションの結果」を指定した場合、以下のように表示されます。「\$2」 は 2 番のアクションの実行結果を意味しています。

セットする値	全て同じフィールドコードをセット	セット先のフィー	ールド
挿入 🖌 = \$2		▶ 作業開始日	日付

関数

カスタマインで利用可能な関数を指定可能です。指定可能な関数はドキュメン ト(関数一覧)のとおりです。

以下の例では、「報告日」フィールドに「today 関数」によって本日の日付を 入力しています。

セットする値	全て同じフィールドコードをセット	セット先のフィー	-ルド
挿入 ❤ = today()		▶ 報告日	日付

3. レコード追加、レコード更新に関する

「やること」一挙解説

想定するアプリと解説の流れ

ここからは実際のカスタマイズを作成しながら各やることの利用例、特徴を解 説します。それぞれの違いをわかりやすくするために共通のアプリ、ほぼ共通 のカスタマイズを作成していきます。

使用するアプリ

『案件管理』アプリ、『活動履歴』アプリ、『発注台帳』アプリを使用します。

アプリ運用の想定とレコード追加・レコード更新の意味

営業担当者が『活動履歴』アプリに活動業務を登録します。登録というのは kintone でいうとレコード追加のことです。以降は「レコード追加」で説明 します。

『活動履歴』アプリにレコードが追加されたとき、まだ『案件管理』アプリ に登録がない場合は、『案件管理』アプリに新規案件としてレコード追加し ます。

『活動履歴』アプリにレコードが追加されたとき、すでに『案件管理』アプリに登録がある場合は、『案件管理』アプリの既存レコードを更新します。 これを以降は「レコード更新」と説明します。



作成するカスタマイズ (3-1~3-9)

3-1 から 3-9 までで作成するカスタマイズは『活動履歴』 アプリまたは 『案 件管理』 アプリに適用します。

『発注台帳』アプリは 3-4、3-5 で作成するカスタマイズの更新先アプリと なっています。

カスタマイズの結果の確認方法

ここでは、レコード追加やレコード更新の結果は、実際のアプリのレコード を見ることによって確認することにします。

では、1つ1つのやることについて解説していきます。

3-1. レコードを追加する

ー番簡単かつシンプルに利用できるのが [やること : レコードを追加する] で す。

[やること:レコードを追加する]は、レコードを追加する[やること]です。 レコードを更新したい場合は、3-2 以降で紹介する別の[やること]を使って ください。設定は、レコードを追加するアプリの指定と、値のセットに関わる マッピング指定の2つだけのため非常にシンプルです。

[やること:レコードを追加する]では必ず1件のレコードが追加されます。

カスタマイズ例

この例では、『活動履歴』アプリにレコードを追加したときに、一緒に『案件管理』 アプリに案件情報を登録します。

『活動履歴』アプリに「案件登録」ボタンを置いておき、ボタンを押したときに『案 件管理』 アプリに登録します。

<図:カスタマイズ>



実行結果

<図:追加のイメージ>

案件管理に登録	レボタンを 新規問い	押す	活動 ^{案件ID・案件4}	履歴ア	アプリ _{在しない}	案件名入力 ? 案件名を入力してください 山田府専取属工事2周
案件ID 顧客ID 101 時度	案件名 顧客名 山田商事株式会	社				マ ****5
C	重点顧客	案件ID		案件名		案件管理アプリ
活動日 こ 2023-08-16 ご 挨拶		001		山田商事	事配線工事2	2期
活動メモ		顧客ID 101		顧客名 山田商事	幕株式会社	
		顧客種別	営業	担当者 新田 蓮		
		最終活動日 2023-08-16	確度 C		金額	案件状況 継続

解説

『活動履歴』アプリにレコード追加したときに、案件が新しく生まれたという ことで『案件管理』アプリにもレコードを追加します。

案件名も決めないといけないので、「案件登録」ボタンを押したときに[やる こと:入力ダイアログを表示する]を使って案件名を入力します。

案件 ID は [やること: 自動採番を行う] を使って自動的に採番しています。

3-2. レコードを更新する(キーの値を直接指定)

カスタマイズ対象のレコードと紐づくレコードを更新する際に使用するのが、 [やること:レコードを更新する(キーの値を直接指定)]です。2-1.レコー ド追加、レコード更新</u>で説明したように、レコード更新をする際には「キー」 が必要になります。レコード更新でパラメーターに指定した「キー」を持つレ コードを探して見つかった場合にレコード更新を行います。指定した「キー」 を持つレコードが見つからない場合には、更新処理は行われません。

[やること: **レコードを更新する(キーの値を直接指定)**] では、「キー」の値に、 表示されているフィールド値をはじめ、他のアクションの結果や固定値などを 指定することができます。

キーを指定してレコード更新をしたい場合は、まず [やること:レコードを更 新する(キーの値を直接指定)] を使うことを考えてみましょう。

カスタマイズ例

ここでは、『活動履歴』アプリのレコードを保存したときに、「案件 ID」で紐 づいている『案件管理』アプリのレコードの「確度」「最終活動日」も更新す る例を紹介します。

「確度」「活動日」は活動履歴アプリに登録しているので、その内容で上書き更 新します。

<図:カスタマイズ>

17	コメント入力				
	レコードを更新する(キーの値を直接指定)		レコードを保存した直後(削除後は	は除く)	
	更新先アプリ	[歩き方][レコード更新]案件管理			
	キーとなるフィールド	案件ID			
	キーの値	=案件ID			
	マッピング	最終活動日=活動日 確度-確度			
	更新の競会をチェックする	9世/又 ⁻ 9世/又 チェックする			

実行結果

<図:更新のイメージ>

		*#D *#4 活動履歴アプリ
		123 山田商事株式会社 販売管理システム導入
		顧客ID 顧客名
		101 山田商事株式会社
		確度 顧客種別
		B 〜 新規開拓 〜
安 (川)		活動日にで保存活動担当
泉叶山 泉叶石	杜ゴ合社 販売管理システル港	2023-07-20 見積提示 ~ 久 久
123 田田岡4	休式去社 敗元官庄シスノム朝	新田 進 ×
顧客ID 顧客名		
101 山田商事	考 案件ID	案件管理アブリ
顧客種別 営業担当者	123	山田商事株式会社 販売管理システム導入
新規開拓 🔷 粕田 舞	顧客ID	顧客名
最終活動日 確度	101	山田商事株式会社
2023-07-16 C	顧客種別	営業相当者
	新規開拓	◎ 粕田 舞
活動履歴の内容で更新	最終活動日 催度	金額 案件状況
	2023-07-20 B	

解説

今回の例では、『活動履歴』アプリの「案件 ID」フィールドと、『案件管理』 アプリの「案件 ID」フィールドをそれぞれのキーとして更新しました。 更新するタイミングは、『活動履歴』アプリのレコード保存後 < 条件:レコー <u>ドを保存した直後(削除後は除く)</u>>としています。こうすると、保存が完了 してから更新する設定となります。『活動履歴』アプリのレコード保存をキャ ンセルした場合は『案件管理』アプリは更新されません。

3-3. レコードを更新または追加する(キーの値を直接指定)

カスタマイズ対象のレコードと紐づくレコードを更新するという点において は、[やること:レコードを更新する(キーの値を直接指定)]と同じですが、 紐づくレコードが更新先アプリに存在しない際にレコード追加ができるのが、 [やること:レコードを更新または追加する(キーの値を直接指定)]です。

カスタマイズ例

3-1. レコードを追加する で『案件管理』アプリに『活動履歴』アプリのレコード内容を元に『案件管理』アプリにレコード追加するボタンをカスタマイズで作成し、

3-2. レコードを更新する(キーの値を直接指定)では『活動履歴』アプリの レコード保存後に『案件管理』アプリの「確度」「最終活動日」を更新するカ スタマイズを作成しました。

[やること: **レコードを更新または追加する(キーの値を直接指定)**]を使うと、 この2つを1つのアクションで実現できます。

<図:カスタマイズ>

24	コメント入力]	
	ボタンをメニュー位置に配置する		詳細画面を表示した時		
	場所 ラベル	レコード詳細メニューの上側 案件管理に登録更新			
	追加位置	一番右に追加する			
27	コメント入力]	
	レコードを更新または追加する(キーの値を直接	妾指定)	ボタンを押した時		
	追加先アプリ	[歩き方][レコード更新]案件管理		ボタン 24	
	キーとなるフィールド	案件ID			
	キーの値	=案件			
	マッピング	案件ID= 案件ID 案件A= 案件名 顧客ID= 顧客ID 顧客名= 顧客名 顧客種別= 顧客種別 営業担当者= 活動担当 最終活動日= 活動日			

実行結果

<図:更新または追加のイメージ>

案件管理	里に登録更新	🛑 ボタン	を押す								
			案件ID		案件名	í.	顧客ID		顧客名	,	
			123		山田商事株式会社 販売管理システム導入		-		山田商	事株式会社	
案件ID	客件ID 客件々		003		アールツー物産 パソコン入れ替え		104		アール	ツー物産	
123		山田商事株:	式会社 販売管理シスラ	008		住井テ	クノロジー Web改修	103		住井テ	ウノロジー
顧客ID		顧客名			1						
101		山田商事株	式会社	案件![ID 案件名		Į.			顧客名
確度	案件ID		案件名		123		山田商事株式会社 販売管理システム導入		101		山田商事株式会社
D	145		1.四商事株式会社	回線工	003		案件IDが存在する時は更認 アールツー物理 ハンコンハルョム	新	104		アールツー物産
	顧客ID		顧客名	145		山田商事株式会社 回線工事 1 期		10			山田商事株式会社
	101 山田商事株式会社			008		案件IDが存在しない時は 住井テクノロシー Web&(1)	追加	103	案條	牛管理アプリ	
	C		重点顧客				活動履歴アプリ				

解説

[やること:レコードを更新する(キーの値を直接指定)] と似ていますが、 違うところは、更新の対象となる、キーで紐づくレコードがなかった場合の動 きです。

2-1. レコード追加、レコード更新で説明したように、レコード更新をする際 には「キー」が必要になります。レコード更新でパラメーターに指定した「キー」 を持つ更新先のアプリのレコードを探して見つかった場合にレコード更新を行 います。ここまでは [やること: レコードを更新する(キーの値を直接指定)] と同じですが、 [やること: レコードを更新または追加する(キーの値を直接 指定)]の場合は、指定した「キー」を持つレコードが見つからなかったときに「レ コード追加」をする点が大きな違いです。

今回の例では、3-2. レコードを更新する(キーの値を直接指定)</u>と同様に『活動履歴』アプリの「案件 ID」フィールドと、『案件管理』アプリの「案件 ID」フィールドをそれぞれのキーとしています。

『活動履歴』アプリの「案件 ID」フィールドに一致する値が『案件管理』アプ リの「案件 ID」フィールドに存在しない場合には、『案件管理』アプリにレコー ド追加をします。

一方、『活動履歴』アプリの「案件 ID」フィールドの値と、『案件管理』アプ

リの「案件 ID」フィールドが一致するレコードが存在する場合には、そのレコードに対してレコード更新を行います。

3-4. レコードを更新する(キーの値をフィールドで指定)

[やること:レコードを更新する(キーの値を直接指定)] と同様にカスタマ イズ対象のレコードと紐づくレコードを更新することができ、キーにフィール ドを指定するのが [やること:レコードを更新する(キーの値をフィールドで 指定)] です。

これは [やること : **レコードを更新する(キーの値を直接指定)**] と似ていま すが、利用できるシーンに一部違いがあります。具体的には、次の例のように テーブル内の値をキー項目として設定した場合に、テーブル行の件数分だけ処 理を繰り返すことができます。

ただし、[やること:レコードを更新する(キーの値を直接指定)]ではキー として「フィールド」「他のアクションの結果」「関数」などが指定可能なの に対して、[やること:レコードを更新する(キーの値をフィールドで指定)] ではキーに指定できるのは「フィールド」のみとなる点に注意が必要です。

カスタマイズ例

この例では、『案件管理』アプリのテーブルの内容をレコードとして書き出し した『発注台帳』アプリを使います。

『案件管理』アプリのテーブルの内容が変わったときに、同時に『発注台帳』 アプリの内容も更新します。

レコード更新では「キー」が必要なため「発注台帳キー」フィールドを用意しています。

<図:カスタマイズ>

7	コメント入力		
	レコードを更新する(キーの値をフィールドで指定)	レコードを保存した直後(削除後は除く)	
	更新先アプリ [歩き方][レコード更]発注台帳	
	キーとなる更新先のフィールド 発注台帳キー		
	キーの値となるこのアプリのフィールド 発注台帳キー		
	発注台帳キー=発注 案件ID-案件ID 発注日=発注日 部品コード=部品コ 政量= 数量	版キー ド	
	更新の競合をチェックする チェックする		

実行結果



■ 受注確定	後記入欄				案件管理	■アプリ			
開始日実績	終了日実績		1	発注					
2023-08-01				発注日	戦略コード	数量	発注台帳キー		削除
発注				2023-08-01	A01	100	001-A01		削除
発注日		数量	発注:	2023-08-01	402	55	001-402		al Re
2023-08-01	A01	100	001	2020 00 01	HUE		OUT AUE		HIBR
2023-08-01	A02	60	00			20	001-B05		削除
2023-08-01	B05	20	001	数量を	変更して保存				
2023-08-01	C10	80	001	2023-00-01	010	77	001-C10		削除
		発注台帳	+-		+-	に紐づく数量	を更新数	r量 案件ID	

			ALL.	2011-
001-A01	2023-08-01	A01	00	001
001-A02	2023-08-01	A02	55	001
001-В05	2023-08-01	B05	20	001
001-C10	2023-08-01	C10	77	001

発注台帳アプリ

解説

このように、[やること:レコードを更新する(キーの値をフィールドで指定)] の《パラメーター:キーの値となるこのアプリのフィールド》にテーブルの中 のフィールドを指定すると、テーブルの件数分だけ処理を繰り返すことができ ます。これは、[やること:レコードを更新する(キーの値を直接指定)]で は実現できないことです。

今回の例ではテーブル行が4行ありますが、1つのアクションだけで4行す べてをキーとして指定し、レコード更新をすることが可能です。

今回の例では、テーブル内のフィールド「発注台帳キー」を「キー」としました。

注意点

この [やること: **レコードを更新する(キーの値をフィールドで指定)**]を一 覧画面で使用する場合、一覧画面に表示された数だけ処理が繰り返されます。 ただし、一覧画面に表示されているレコードだけが対象になりますので注意が 必要です。一覧画面の条件に合致していても、表示されていない次ページ以降 のレコードは処理対象になりません。

次ページ以降のレコードも含めた処理を行いたい場合は、これ以降の節で紹介 する他のやることを使用してください。



3-5. レコードを更新または追加する(キーの値をフィー

ルドで指定)

カスタマイズ対象のレコードと紐づくレコードを更新するという点において は、[やること:レコードを更新する(キーの値を直接指定)]と同じですが、 紐づくレコードが更新先アプリに存在しない際にレコード追加ができるのが、 [やること:レコードを更新または追加する(キーの値をフィールドで指定)] です。

これは [やること : **レコードを更新または追加する(キーの値を直接指定)**] と似ていますが、利用できるシーンに一部違いがあります。具体的には、次の 例のようにテーブル内の値をキー項目として設定した場合に、テーブル行の件 数分だけ処理を繰り返すことができます。

ただし、[やること:レコードを更新または追加する(キーの値を直接指定)] ではキーとして「フィールド」「他のアクションの結果」「関数」などが指定可 能なのに対して、[やること:レコードを更新または追加する(キーの値をフィー ルドで指定)]ではキーに指定できるのは「フィールド」のみとなる点に注意 が必要です。

カスタマイズ例

<u>3-4. レコードを更新する(キーと値をフィールドで更新)</u>でテーブルを書き 出したレコードに対して、変更内容を更新する処理を作りましたが、この方法 ではあとから追加したテーブル行に対しては対応ができません。

[やること:レコードを更新または追加する(キーの値をフィールドで指定)] を使ったカスタマイズでは次の例のようになります。

<図:カスタマイズ>

8	コメント入力		
	レコードを更新または追加する(キーの値をフィール	ドで指定)	レコードを保存した直後(削除後は除く)
	追加先アプリ	[歩き方][レコード更新]発注台帳	現在の画面が追加画面または編集画面ならば
	キーとなる更新先のフィールド	発注台帳キー	
	キーの値となるこのアプリのフィールド	発注台帳キー	
	マッピング	発注台帳キー= 発注台帳キー 案件ID-案件ID 発注日=発注日 部品コード=部品コード 数量=数量	

実行結果

<図:更新または追加のイメージ>

						案件	管	理アプリ				
発注日	部品コ-	- K		数量		発注台帳キー						
2023-08-01	A01				120	001-A01	_					
2023-08-01	A02				55	001-A02						
2023-08-01	B05				20	001-B05						
2023-08-01	C10				77	001-C10						
2023-08-03	E14				10	001-E14						
2023-08-03	F03				20	001-F03						更新
							7					1
発注台帳キー		発注日	部品コー	1	数量	案件ID		発注台帳キー	発注日	部品コード	\$. A	案件ID
001-A01		2023-08-01	A01		100	001		001-A01	2023-08-01	A01	120	001
001-A02		2023-08-01	A02	Г	55	001		001-A02	2023-08-01	A02	55	001
001-B05		2023-08-01	B05		20	001		001-B05	2023-08-01	B05	20	001
001-C10		2023-08-01	C10		77	001		001-C10	2023-08-01	C10	77	001
発注台	帳	アプリ						001-E14	2023-08-03	E14	10	001
						2 行追加	~	001-F03	2023-08-03	F03	20	001

解説

[やること:レコードを更新する(キーの値をフィールドで指定)] と似てい ますが、違うところは、更新の対象となるレコードがなかった場合の動きです。 2-1.レコード追加、レコード更新</u>で説明したように、レコード更新をする際 には「キー」が必要になります。レコード更新でパラメーターに指定した「キー」 を持つ更新先アプリのレコードを探して、レコードが見つかった場合にレコー ド更新を行います。ここまでは [やること:レコードを更新する(キーの値を フィールドで指定)] と同じなのですが、[やること:レコードを更新または 追加する(キーの値をフィールドで指定)] の場合は、指定した「キー」を持 つレコードが見つからなかったときに「レコード追加」をする点が大きな違い です。

今回の例では、すでに登録されていたテーブル行に対しては変更内容をレコー ド更新、新しく追加されたテーブル行に対してはレコード追加しています。レ コードがすでに登録されているかどうかは、「キー」で判断します。

3-6. レコードを書き出す

[やること:レコードを更新する(キーの値を直接指定)]や[やること:レコー ドを更新または追加する(キーの値をフィールドで指定)]と同様にカスタマ イズ対象のレコードと紐づくレコードを更新、または追加することができるの が、[やること:レコードを書き出す]です。《パラメーター:レコード》で元 になるレコードを指定するため、複数レコードを対象にでき、そのレコードの 値を元にレコードを更新することが可能です。

なお、カスタマイズ対象のアプリと同じアプリに対して [やること : レコード <u>を書き出す</u>] を使う場合は、レコード更新・レコード追加のどちらも行うこと ができますが、違うアプリに対して [やること : レコードを書き出す] を使う 場合は、レコード追加のみ可能です。

カスタマイズ例

この例では、『案件管理』アプリの一覧画面でチェックをつけた案件に対して、 営業担当者をログインユーザーに一括更新します。

<図:カスタマイズ>

1	コメント入力	
	一覧にチェックボックス列を追加する	一覧画面を表示した時
2	コメント入力	
	ボタンをメニュー位置に配置する	一覧画面を表示した時
	場所 一覧画面メニューの右側	
	ラベル 案件担当変更	
	追加位置 一番右に追加する	
4	コメント入力	
	一覧で選択されたレコードを取得する	ボタンを押した時
		ボタン 2
6	באאראל	
	ログインユーザーを取得する	他のアクションの実行が完了した時
	セット先フィールド(省略可) 〈選択されていません〉	アクション 4
3	コメント入力	
	確認ダイアログを表示する	他のアクションの実行が完了した時
	チェックを付けた案件について、	アクション 6
	担当をあなたに変更します。	
	キャンセルボタンの名前 キャンセル	
5	コメント入力	
	レコードを書き出す	確認・入力ダイアログで「OK」を押した時
	レコード 4	確認・入力ダイアログアクション 3
	書き出し先アプリ [歩き方][レコード更新]案件管理	
	マッピング 営業担当者= \$6	
	更新または追加 既存レコードを更新	

実行結果

<図:更新のイメージ>

					案件管理ア	プリ
案件ID	案件名	営業担当者		案件ID	案件名	営業担当者
001	山田商事配線工事2期	▲ 新田 蓮		001	山田商事配線工事2期	▲新田 蓮
123	山田商事株式会社 販売管理システム導入	☆ 粕田 舞		123	山田商事株式会社 販売管理システム…	1 新田 蓮
003	アールツー物産 パソコン入れ替え	☆ 粕田 舞		003	アールツー物産 パソコン入れ替え	▲ 新田 蓮
008	住井テクノロジー Web改修	☆ 粕田 舞	-	008	住井テクノロジー Web改修	▲新田 蓮
005	城不動産複合機提案	☆ 粕田 舞	J	005	城不動產複合機提案	▲ 新田 蓮

解説

レコード更新には「キー」が必要なことは 2-1.レコード追加、レコード更新 で説明していますが、今回の [やること:レコードを書き出す]では「キー」 は指定しません。これはなぜかというと、[やること:レコードを書き出す] でレコード更新をするときはレコード番号を「キー」にして更新が行われてい るからです。このため、パラメーターの「既存レコードを更新」は更新先アプ リが元になるレコードのアプリと同じ場合にのみ指定が可能です。

今回の例では、チェックを付けたレコードに対して更新をしました。このよう に元になるレコードに対して更新したり、元になるレコード全件を対象にレ コード追加をする場合には、[やること:レコードを書き出す]を使うと処理 速度も速く、APIリクエスト数も少なく済むのでおすすめです。

今回の例では、[やること:レコードを書き出す]を使ってレコードの更新を 行いましたが、パラメーターに「新規にレコードを追加」を選択することでレ コード追加を行うこともできます。なお、「新規にレコードを追加」の場合は、 レコードを追加する先のアプリが元になるレコードと同じアプリでも違うアプ リでも実行する事ができます。

3-7. レコードを1つずつ書き出す

[やること:レコードを書き出す]と同様に複数レコードを対象にレコードを 追加・更新することが可能なのが[やること:レコードを1つずつ書き出す] です。

[やること:レコードを書き出す]との違いは、1レコードずつ書き出し処理 を行うため、kintone の Webhook 通知の送信が可能な点です。それ以外の 機能は [やること:レコードを書き出す]と同じですので、以下の利用シーン 以外の場合は [やること:レコードを1つずつ書き出す]ではなく [やること: レコードを書き出す]を選択するようにしましょう。

利用シーン

kintone の Webhook 通知の送信ができると、どういった事が実現できるのでしょうか?

たとえば、次のような利用シーンが考えられます。

- ・kintone のレコード追加をきっかけとして連携サービスを利用したいとき。
- ・kintone の Webhook 通知の送信を利用して、他システムにデータ連携 したいとき。
- ・カスタマインの kintone アプリの Webhook のカスタマイズを動かした いとき。

ほかにも、kintone の Webhook 通知の送信を利用した処理を動かしたいと きが、利用シーンとして考えられます。

カスタマイズ例

3-6. レコードを書き出す の説明で、チェックを付けたレコードに対して、担 当者を一括変更する例を紹介しました。この方法だとレコードの一括更新がス ムーズに行えるのですが、Webhookの通知が送信されません。 こういった時に [やること: レコードを1つずつ書き出す]を利用すると、一 度にまとめて更新しつつ、1 件毎に Webhook 通知を送信することができます。

<図:カスタマイズ>

10	コメント入力	
	一覧にチェックポックス列を追加する	一覧画面を表示した時
_		
17	コメント入力	
	ボタンをメニュー位置に配置する	一覧画面を表示した時
	場所 一覧画面メニューの右側	
	ラベル 案件担当変更(1つずつ)	
	追加位置 一番右に追加する	
_		
18	コメント入力	
	一覧で選択されたレコードを取得する	ボタンを押した時
		ボタン 17
19	- コメント入力	
	ログインユーザーを取得する	他のアクションの実行が完了した時
	セット先フィールド(省略可) 〈選択されていません〉	アクション 18
20	コメント入力	
	確認ダイアログを表示する	他のアクションの実行が完了した時
	確認ダイアログを表示する メッセージ入力	他のアクションの実行が完了した時 アクション 19
	確認ダイアログを表示する メッセージ入力 チェックを付けた案件について、 担当をあなたに変更します。 のKボダンの条節 のK	他のアクションの実行が完了した時 アクション 19
	確認ダイアログを表示する メッセージ入力 チェックを付けた案件について、 担当をあなたに変更します。 のKボタンの名前 OK キャンセルボタンの名前 キャンセル	他のアクションの実行が完了した時 アクション 19
	確認ダイアログを表示する メッセージ入力 チェックを付けた案件について、 担当をあなたに変更します。 のKボタンの名前 のK キャンセルボタンの名前 キャンセル	他のアクションの実行が完了した時 アクション 19
21	確認ダイアログを表示する メッセージ入力 チェックを付けた案件について、 担当をあなたに変更します。 のKボタンの名前 のK キャンセルボタンの名前 キャンセル コメント入力	他のアクションの実行が完了した時 アクション 19
21	確認ダイアログを表示する メッセージ入力 チェックを付けた案件について、 担当をあなたに変更します。 のKボタンの名前 キャンセルボタンの名前 キャンセル コメント入力 レコードを1つずつ書き出す	他のアクションの実行が完了した時 アクション 19 確認・入力ダイアログで「OK」を押した時
21	確認ダイアログを表示する メッセージ入力 チェックを付けた案件について、 担当をあなたに変更します。 のKボタンの名前 OK キャンセルボタンの名前 キャンセル コメント入力 レコードを1つずつ書き出す レコード 18	他のアクションの実行が完了した時 アクション 19 確認・入力ダイアログで「OK」を押した時 確認・入力ダイアログで「OK」を押した時
21	確認ダイアログを表示する メッセージ入力 チェックを付けた案件について、 担当をあなたに変更します。 のKボタンの名前 ペ キャンセルボタンの名前 キャンセル コメント入力 レコードを1つずつ書き出す レコード 18 書き出し先アブリ [歩き方][レコード更新)案件管理	他のアクションの実行が完了した時 アクション 19 確認・入力ダイアログで「OK」を押した時 確認・入力ダイアログで「OK」を押した時

実行結果

<図:更新のイメージ>

						案件管理ア	アプリ
	案件ID	案件名	営業担当者		案件ID	案件名	営業担当者
	001	山田商事配線工事2期	▲ 新田 蓮		001	山田商事配線工事 2 期	▲新田 蓮
•	123	山田商事株式会社 販売管理システム導入	☆ 粕田 舞		123	山田商事株式会社 販売管理システム…	1 新田 蓮
	003	アールツー物産 パソコン入れ替え	☆ 粕田 舞		003	アールツー物産 パソコン入れ替え	1 新田 蓮
•	008	住井テクノロジー Web改修	☆ 粕田 舞	-	008	住井テクノロジー Web改修	1 新田 蓮
	005	城不動産複合機提案	☆ 粕田 舞	J	005	城不動產複合機提案	▲ 新田 蓮

このカスタマイズを動かした kintone レコードの更新結果としては [やるこ と:<mark>レコードを書き出す</mark>] と同じになります。

違いは Webhook 通知の送信が起きる点です。

解説

[やること:レコードを書き出す] に似ていますが、[やること:レコードを書 き出す] はレコードをまとめて処理するのに対し、[やること:レコードを1 つずつ書き出す] では1レコードずつ処理する点が違います。そのため、扱う レコード件数が多い場合は kintone API リクエスト数が多くなりやすく、処 理時間も長くなりやすいです。

原則として、[やること:**レコードを書き出す**]を使用するようにし、 Webhook 通知の送信を作動させたい場合にのみ [やること:**レコードを1つ ずつ書き出す**] を選ぶようにしましょう。

注意点

kintone の仕様により Webhook の通知は、ドメイン単位で1分間に 60 回ま で送信されます。1 分間に 60 回を超えてレコードを操作すると、61 回目以 降の操作では Webhook の通知が送信されません。

この制限はアプリ単位ではなくドメイン単位での制限であることは意識してお きましょう。

3-8. レコードをもとに別のレコードを更新する

複数のレコードをもとにレコードを更新することに関しては [やること : <u>レ</u> <u>**コードを書き出す**</u>] と同じですが、更新先アプリがもとになるレコードのアプ リと別のアプリであっても更新が可能な点が違います。

[やること:レコードをもとに別のレコードを更新する]は、[やること:レコー ドを書き出す]に比べると処理時間が長くなりやすく、APIリクエスト数も多 くなります。同じアプリに対する更新の場合は、[やること:レコードを書き 出す]を利用することをおすすめします。

一方、違うアプリに対して更新する場合には [やること : レコードを書き出す] では実現できませんので、[やること : レコードをもとに別のレコードを更新 する] を利用してください。

カスタマイズ例

この例では、『活動履歴』アプリの一覧画面で選択したレコードを元に、『案件 管理』アプリの「案件状況」フィールドの内容を更新するカスタマイズを作り ました。

<図:カスタマイズ>

20	コメント入力		
	一覧にチェックボックス列を追加する		一覧画面を表示した時
			現在の一覧が指定の一覧ならば
			一覧の名前 一覧2
21	コメント入力		
	ボタンをメニュー位置に配置する		一覧画面を表示した時
	場所	一覧画面メニューの右側	現在の一覧が指定の一覧ならば
	ラベル	案件状況更新	- 覧の名前 一覧2
	追加位置	一番右に追加する	
23	742614		
	選択肢から選択するダイアロクを表示する		ホタンを押した時
	メッセージ入力	更新後の案件状況を選択してくだ さい	ホタン 21
		受注	
	選択肢	※続 なみ ず	
	未選択を表す「」	追加する	
	OKボタンの名前	ОК	
	キャンセルボタンの名前	キャンセル	
	未選択を許可するかどうか	未選択でもOKを許可する	
	初期値	〈入力されていません〉	
22			
	一覧で選択されたレコードを取得する		確認・入力ダイアログで「OK」を押した時
			確認・入力ダイアログアクション 23
22			
52			
	レコードをもとに別のレコードを更新する		他のアクションの実行が完了した時
	更新先アプリ	[歩き方][レコード更新]案件管理	アクション 22
	キーとなる更新先のフィールド 元になるレコード	22	
	キーの値となる元になるレコードのフィールド	案件ID	
	マッピング	案件状況=\$23	
	更新の競合をチェックする	チェックする	

実行結果

<図:更新のイメージ>

-	覧2	\sim	°℃ ~	7	案	件状況更新					
ボタンを押して「受注」を選ぶ											
	活動日	活動内容	活動担当	案件ID	THE OWNER OF	案件名					
	2023-07-20	見積提示	▲ 新田 蓮	123	23 山田商事株式会社 販売管理シス			(?)			
•	2023-07-07	ご挨拶	■ 新田 蓮 009 賀寿丸建設 PCセットアップ作業					更新後の案件状況を選択してください			
「案件ID」で紐づくレコードを更新 案件ID ア 代況 案件ID ア 代況 案件A 009 継続 貿寿丸建設 PCセットアップ作業											
U	123	継続	4	DAY	案件ID	案件状況	案件名	案件管理アプリ			
				C	009	受注	賀寿丸建設	PCセットアップ作業			
			1	1	23	受注	山田商事株	式会社 販売管理システム導入			

解説

この例では『活動履歴』アプリの一覧画面で選択したレコードを元に、『案件 管理』アプリを更新しました。

今回の例では「キーとなる更新先のフィールド」には「案件 ID」フィールド を指定しています。一覧で選択した複数レコードの「案件 ID」が同じレコー ドに対して、一括で『案件管理』アプリの「案件状況」フィールドの内容を更 新します。

「案件状況」の値は、[やること:<mark>選択肢から選択するダイアログを表示する</mark>] を使って、プルダウン形式で選択しています。

[やること:レコードを書き出す]の場合は、元になるレコードと違うアプリ に対してレコード更新することができませんでしたが、[やること:レコード をもとに別のレコードを更新する]を使うと、それが可能になります。

元になるレコードを指定するために、[やること:レコードをもとに別のレコー <u>ドを更新する</u>]のアクションの前にレコードを取得する[やること]([やるこ と:一覧の条件でレコードを全件取得する]など)を使ったアクションを実行 しておく必要があります。

コラム:1件ずつ処理することが適しているケース

[やること:レコードをもとに別のレコードを更新する]のアクションでは、1件ずつ更新先のレコードを取得したあとにレコード更新を行っており、処理時間とAPIリクエスト数が多くなりやすいです。 一方でメリットもあり、1件ずつ更新先のレコードを取得することで、下図のようにマッピングで更新先のレコードの値を使うことができます。

たとえば、在庫数の管理をしている『備品管理アプリ』と入出庫数を記 入する『入出庫記入アプリ』を例に説明します。以下の例だと、すでに『備 品管理アプリ』のA4ノートは 25 個の在庫があります。そのため、入 庫があった際には入庫数の5を『備品管理アプリ』に登録するのではな く、すでに在庫数登録している 25 個に入庫数の5 個を足す必要があり ます。



[やること:レコードをもとに別のレコードを更新する]が実行される と、「製品コード」のフィールド値が同じレコードを『備品管理アプリ』 から1つ取得します。取得したレコードの値はマッピングで使うことが できるため「更新後の在庫数 = 更新前の在庫数 + 入出庫数」のように、

図:在庫	数計算の例>					
Ĵ フィールド選択([レコ−	- ド更新]備品管理)					
登切したフィールドをずに話	etal = -					
マッピング元アプリ	マッピング先アプリ カスタマイ	ズ対象アプリ 検索 30 フ・	ィールド選択 ([レコード更	[新]入出庫記入)		
フィールド名	フィールドコード テーブ	ル フィー				
○ 製品コード	製品 ド	選択した 文字列	:フィールドを式に追加し	ます。		
○ 品名	品名	文字列	ッピング元アプリ	マッピング先アプリ	カスタマイズ対象アプリ	検索
 在庫数 	在庫数		フィールド名	フィールドコード	テーブル	フィールドタイプ
0						大子列 (11)/
			¹²	品名		文字列(1行)
			、出庫数	入出庫数		数值
				-		
	J7-) C79E20	([レコード更新]頒品管理)				
	更新前の備品管理ア	プリの在庫数 + 入出庫	記入アプリの入出	庫数		
	挿入 🗸			ド 文字	列 (1行)	
		/			// (413/	
	挿入 ~		> 品名	文字	列(1行)	
	挿入 🖌 = @out.在原	重数+入出庫数	> 在庫	遨 数值		

上図のように「マッピング先アプリ」からフィールドを選ぶと「@out.」 がつき、更新先アプリのフィールド値を使用できます。また、下図のよ うに「カスタマイズ対象アプリ」からフィールドを選ぶと「@this.」が フィールドコードの前に付き、カスタマイズ対象アプリのフィールド値 を使用できます。

<図:マッピング>

#入 ~ = 案件名		- ∰入 ~ = @out.案件名	挿入 → =@this.案件名
	選択したフィールドを式に追加	します。	
	マッピング元アプリ	マッピング先アプリ カスタマイズ対象アプリ	検索
	フィールド名	フィールドコード テーフル	フィールドタイプ
	○ 案件IDルックアップ	案件IDルックアップ	ルックアップ(文字列)
	○ 案件名	案件名	文字列(1行)
	 顧客IDルックアップ 	顧客IDルックアップ	ルックアップ(文字列)
	○ 顧客名	顧客名	文字列(1行)
	@ 確度	確度	ドロップダウン
	○ 顧客種別	顧客種別	ドロップダウン
	○ 活動日	活動日	日付
	○ 活動内容	活動内容	ドロップダウン
	○ 活動担当	活動担当	ユーザー選択

3-9. レコードをもとに別のレコードを更新または追加する

複数のレコードをもとにレコードを更新する点に関しては、[やること:レコー <u>ドをもとに別のレコードを更新する</u>]と同じですが、紐づくレコードが更新先 アプリに存在しない際にレコード追加ができるのが[やること:レコードを とに別のレコードを更新または追加する]です。

カスタマイズ例

3-8. レコードをもとに別のレコードを更新する 画面で選択したレコードを元に、『案件管理』アプリの「案件状況」フィール ドの内容を更新するカスタマイズを作りましたが、これでは『案件管理』アプ リに登録がない案件は更新されません。

これに対し、[やること:レコードをもとに別のレコードを更新または追加する] を使ったカスタマイズだと、『案件管理』アプリに登録がある案件については レコード更新、『案件管理』アプリに登録がない案件についてはレコード追加 を行います。

<図:カスタマイズ>

34	コメント入力			
	一覧にチェックボックス列を追加する		1	一覧画面を表示した時
			-	
35	コメント入力			
	ボタンをメニュー位置に配置する]	一覧画面を表示した時
	場所	一覧画面メニューの右側		
	ラベル	案件状況 更新または追加		
	追加位置	一番右に追加する		
36	コメント入力		_	
	選択肢から選択するダイアログを表示する			ボタンを押した時
	メッセージ入力	更新後の案件状況を選択してくだ		ボタン 35
		受注		
	選択肢	失注 継続		
		クローズ		
	未選択を表す「」	追加する		
	OKボタンの名前 キャンセルボタンの名前			
	キャンセルホランの名前	未選択でもOKを許可する		
	初期値	〈入力されていません〉		
37	コメント入力			
	一覧で選択されたレコードを取得する		1	確認・入力ダイアログで「OK」を押した時
				確認・入力ダイアログアクション 36
38	コメント入力			
	レコードをもとに別のレコードを更新または追	加する]	他のアクションの実行が完了した時
	追加先アプリ	[歩き方][レコード更新]案件管理		アクション 37
	キーとなる更新先のフィールド	案件ID		
	元になるレコード	37		
	キーの値となる元になるレコードのフィールド	案件ID		
		顧客ID= 顧客ID 顧客名= 顧客名		
	マッピング	営業担当者= 活動担当 最終活動日= 活動日		
		確度=確度 案件状況= \$36		

実行結果

<図:更新・追加のイメージ>



解説

[やること:レコードをもとに別のレコードを更新する]と似ていますが、違うところは、更新の対象となるレコードがなかった場合の動きです。

この例では、3-8.レコードをもとに別のレコードを更新すると同様に『活動 履歴』アプリの一覧画面で選択したレコードを元に、『案件管理』アプリを更 新しました。さらに、《パラメーター:キーの値となる元になるレコードのフィー ルド》である「案件 ID」に一致する案件 ID が『案件管理』アプリに存在しな い場合には、レコード追加が行われます。

[やること:レコードをもとに別のレコードを更新する]と同じく、元になる レコードを指定するために、[やること:レコードをもとに別のレコードを更 新または追加する]のアクションの前にレコードを取得する「やること」(一 <u>覧の条件でレコードを全件取得するなど</u>)を使ったアクションを実行しておく 必要があります。 [やること:レコードをもとに別のレコードを更新または追加する]は、[やる こと:レコードをもとに別のレコードを更新する]と同様に、他の更新・追加 系のやることに比べると処理時間が長くなりやすく、使用する API リクエス ト数も多くなります。

注意点

更新元レコードのキーの値《パラメーター:キーの値となる元になるレコード のフィールド》が空のレコードが存在する場合には注意が必要です。 キーの値が空のレコードが複数存在する場合すべてキーが同一であるとみなさ れ、同じレコードを繰り返し更新し、最後に更新された値のみが有効となりま す。

具体的には以下のような動きとなります。

								活動層	夏歴ア	プリ	J		
活動日	活動内容	活動担当	案件ID	案件名		顧客ID	1	顧客名					
2023-09-13	商談	☆ 粕田 参		モバイルアプリ開発案件				城酒造		10	•		
2023-09-11	セミナー	☆ 粕田 参	555	kintone アプリ開発		999		粕玉エンターフ	ライズ	1.0			
2023-09-08	問い合わせ	☆ 粕田 参		SFA/CRM 導入案件		_		アールツー物産	ł	10			
どちらもキーが空白(キーが重複)											案	件管理ア	プリ
案件ID	案件	⊧状況	案件名		営業打	包当者		客ID	顧客名			最終活動日	確度
	継続	5	SFA/CRM	導入案件	会粕		3!	50	アールツ	一物産		2023-09-08	в
555	受注		kintone フ	アプリ開発	+-7	が重複し	.τ	いるため、	同じレコ	コードを	を更新する)23-09-11	В

上記のような動作を防ぐためには、原則としてキーとなるフィールドは重複禁 止、かつ必須に設定することが望ましいです。

4.「やること」使い分けのポイント

本冊子ではレコード更新に関する多くの「やること」を紹介しました。さまざ まな「やること」の利用例や注意点を読んでも、「結局どれを使ったら良いの?」 と悩まれることもあるかと思います。ここでは、そんな方のためにフローチャー トを用意しました。

初めての方にも理解していただきやすいよう、詳細な説明は省いております。 要件によっては一部合致しないケースもありますが、そこはご理解のうえ簡易 的なチャートとしてご利用ください。「レコードを追加する」は用途がシンプ ルで悩まれるケースがほとんど無いと考えられるため、以下の図からは省略し ています。



また、各「やること」ごとに実現できること・実現できないことを一覧として まとめました。上記のフローチャートと合わせてご活用ください

	対象レコ	ード件数	更新テ アプリ	むと同じ への処理	更新元 アプリイ	本書での	
やること名	更新元	更新先	追加	更新	追加	更新	補け
<u>レコードを更新する</u> (キーの値を直接指定)	1	1 or 多	х	0	х	0	<u>3-2</u>
レコードを更新または追加する (キーの値を直接指定)	1	1 or 多	0	0	0	0	<u>3-3</u>
レコードを更新する (キーの値をフィールドで指定)	1	1 or 多	х	0	х	0	<u>3-4</u>
<u>レコードを更新または追加する</u> (キーの値をフィールドで指定)	1	1 or 多	0	0	0	0	<u>3-5</u>
<u>レコードを書き出す</u>	多	多	0	0	0	Х	<u>3-6</u>
<u>レコードを1つずつ書き出す</u>	多	多	0	0	0	х	<u>3-7</u>
レコードをもとに別のレコードを <u>更新する</u>	多	多	х	0	Х	0	<u>3-8</u>
<u>レコードをもとに別のレコードを</u> 更新または追加する	多	多	0	0	0	0	<u>3-9</u>

5. さらにアプリを便利に活用するために

ここまでの例を参考にカスタマイズをして運用してみると、当初は想定してい なかったパターンに気がつくかもしれません。

また、本冊子は「レコード更新」の説明に重点を置いているため、実運用で利 用するために必要となる他の設定は省いている部分があります。

以下にいくつかの例を紹介します。

キーが空欄のままレコード追加しないための対策

3-3. レコードを更新または追加する(キーの値を直接指定)の例では、『活動 履歴』アプリの「案件 ID」フィールドが空欄の場合に『案件管理』アプリにレコー ド追加をすると、「案件 ID」フィールドが空欄の状態のレコードとなってしま います。「案件 ID」フィールドはキーとして使いたいので、空欄ではなく、重 複しない値を入れておきたいです。

こんな時には、レコード追加する際に [やること:<u>自動採番を行う</u>] で採番を 行うのも l つの方法です。

また、<条件:ボタンを押した時 > に「案件 ID」フィールドが空欄かどうかで、 空欄であれば <u>3-1. レコードを追加する</u>、空欄でなければ <u>3-2. レコードを更</u> <u>新する(キーの値を直接指定)</u>に分岐するという方法もあります。

テーブル行を削除したいときの対策

3-4. レコードを更新する(キーの値をフィールドで指定)、3-5. レコードを 更新または追加する(キーの値をフィールドで指定)の例では、テーブル行の 内容を別アプリのレコードに追加や更新をしました。ただ、テーブル行は行追 加だけではなく、「-」ボタンを押すことで削除することも可能です。

『案件管理』アプリのテーブル行を削除した際、連携している『発注台帳』ア プリのレコードも削除したいところなのですが、カスタマインにはレコードを 削除する [やること] はありません。そのため、[やること:テーブルの行増 減ボタンを非表示にする] を使ってテーブル行の削除ボタンは無効とし、その 代わりに該当のテーブル行が無効であることを示す「削除チェックボックス」 を使うことをおすすめします。

詳しくは、サポートサイト内「<u>テーブルを別アプリのレコードに書き出し、常</u> <u>に同期を取る方法</u>」をご覧ください。 ここに紹介しているのは一部の例です。運用していて、作ったカスタマイズで は足りないパターンを発見したら、どのようなカスタマイズを作れば対応がで きるのか、考えてみましょう。

いろいろな方法があると思いますので、動作を試しながら、運用に合ったカス タマイズを検討してみてください。

6. うまく動かないときは?

この章では、うまく動かない時に原因を見つけるための方法と、便利なカスタ マインの機能について紹介します。

6-1. カスタマイズは少し作っては確認、

少し作っては確認の繰り返しで育てよう

カスタマインのカスタマイズは複数のアクションを組み合わせて作成すること が多く、慣れてくると多くのアクションによって構成される大きなカスタマイ ズを、一度も kintone アプリで試して動かさずに作ってしまいがちです。 このようにして一度も実行結果を試さずに作成した大きなカスタマイズは、う まく動かなかった場合にどのアクションに原因があるかを見つけるのが非常に 大変なことになってしまいます。

複数のアクションを組み合わせたカスタマイズでは、少し作ったら実行して動 きを確認する→問題があれば直す→動きを確認する→問題なければ次のカスタ マイズを追加する、という手順の繰り返しで作成することをおすすめします。

6-2. うまく動かない時に最初に確認するポイント

カスタマイズを「kintone アプリへ登録」して実行してみると、エラーが出た り、想定とは異なる結果になったりと、うまく動かないことがよくあります。 うまく動かないときは、ひとつひとつ原因となりそうな箇所を確認し、どこに 原因があるのかを絞り込んでいくことになります。

カスタマインのほかに JavaScript やプラグインが入っている時

JavaScript やプラグインは、カスタマインのカスタマイズの動作に影響を与 える場合があります。

そのためカスタマイズがうまく動かないときは、まずは JavaScript やプラグ インを削除・無効化し、kintone アプリにカスタマインのカスタマイズのみが 設定されている状態にして動作を確認してみてください。これは、原因がカス タマインのカスタマイズにあるのか、それとも JavaScript やプラグインが影 響して動かないのかを判別し、問題を切り分けるために行います。 kintone アプリにカスタマインのカスタマイズのみが登録されている状態で問題なく動作するのであれば、JavaScript やプラグインが影響してカスタマイズが動いていない可能性が高いです。

逆に、JavaScript やプラグインを削除・無効化してもカスタマイズがうまく 動かない時はカスタマインのカスタマイズが誤っている可能性が高いです。カ スタマイズを見直してみてください。

※ドキュメントの「<u>外部サービス連携</u>」で連携している以外のプラグイン、及び自作 JavaScript ファイルが含まれた状態では、カスタマインの動作は保証 外となります。

外部サービス連携:

https://docs-customine.gusuku.io/ja/actions/external/

(ドキュメントで「外部サービス連携」を御覧ください)

※ JavaScript は後でもとに戻せるように、削除する前にダウンロードするな どして保存しておいてください。

6-3. エラーが表示されたとき

エラーが発生した際に表示されるメッセージには、原因を究明するための情報 が含まれています。エラーダイアログが表示されると驚いてしまう事もあると 思いますが、エラーの原因がそのまま記載されている事もありますので、一度 心を落ち着けて読んでみてください。

カスタマインのカスタマイズ画面で表示されるエラー

カスタマインのカスタマイズ画面で表示されるエラーです。

下記の例では「ページ (3): アクション (11): アクション: 無効なアクションで す。(10)」とあります。

冒頭の「ページ (3): アクション (11) :」の部分がエラーとなっているアクション、その後がエラーの内容を示しています。

今回はアクション番号 11 で < 条件:他のアクションの実行が完了した時 > に 指定されたアクションが使用できない、というエラーです。

カスタマイズを見てみますと、アクション番号 10 が無効化されているため、 使用できない状態になっていることがわかります。

~-	ジ(3):アクション(11):アクション制約なアクションです。(10)			×
10	382+3.8			•••
	ボタンをメニュー位置に配置する	0	一覧画賞を表示した時	0
	 編展 一覧目的メニューの下例 ラベル まとめてセット 追加装置 一番もに追加する 		条件を追加	
11	382133			80 ^ ···
	フィールドに値をセットする	0	他のアクションの実行が完了した時	0
	フィールド	7	7953> 10	
			フィールド値が特定の値ならば	😵 🕲 📀
			2メールド 副宮コード	
			条件 なしい	
			比較値 (八力されて	いません)
			条件を追加	

カスタマイズ画面に表示されるエラーが解消するまでは「kintone アプリへ登録」を行うことができません。メッセージの内容を確認して、アクションの設定を確認・修正してみてください。

kintone アプリで表示されるエラーダイアログ

このような白いダイアログは、カスタマインによって表示されるダイアログで す。こういったダイアログ表示は、カスタマインのカスタマイズでやることや 条件の使い方が誤っている場合に表示されるエラーメッセージです。



このエラーは、[やること: <u>ルックアップを取得しなおす</u>]を、レコード保存前・ 保存後のタイミングで実行した時に表示されるエラーです。

エラーメッセージの前にある(25)のように、() で囲まれた数字が記載されている場合は、このアクション番号のアクションでエラーが起きている事を 意味しています。今回の例では(25)ですので、【アクション:25】のアクショ ンでエラーが起きていることがわかります。

コメント入力	400 🔨 🕂
ルックアップを取得しなおす	 レコードを保存する直筋(削除時は除く)
ルックアップフィールド ホックアップ	条件を追加

このようなエラーが表示されたら、<u>ドキュメント</u>の「制限事項」の項目を確認 してみてください。

[やること:<u>ルックアップを取得しなおす</u>]の制限事項は下記のように記載されています。

制限事項 レコード編集画面、レコード追加画面のみで動作します。それ以外の画面で使用するとエラーが表示されます。 • レコードを保存するタイミングでは動作しません。レコードを保存するタイミングで「ルックアップを取得しな おす」した場合の動作は以下のようになります。 。バージョン1.131(2021年5月13日リリース)以前の場合、エラーは起こりませんが、ルックアップの取得は行 われません。 ○ バージョン1.132(2021年5月20日リリース)以降の場合、エラーが発生します。保存処理は停止しません。 ルックアップの現在値から、ルックアップの検索結果が1件に決まらない場合は、ルックアップの取得は行われ ません。

今回のエラーは、[やること:<u>ルックアップを取得しなおす</u>]が、レコードを 保存するタイミングの < 条件:レコードを保存する直前(削除時は除く) > で は使用できないため起こりました。使用するやることを変更するか、条件を変 える必要があります。

次のエラーは先ほどのエラーダイアログと同様の白いダイアログですが、こち らのメッセージはカスタマインでカスタマイズを実行した結果、kintone から エラーが返ってきた場合のダイアログです。



見分け方は [kintone からの応答] というメッセージが記載されているかどう かです。[kintone からの応答] とあれば、kintone からエラーが返ってきた 場合のダイアログであることがわかります。

この場合は、kintone 側からエラーが返ってきているため、kintone のプロセ ス管理の設定とカスタマイズで指定したステータスの整合性が取れていない、 カスタマイズを実行したユーザーが kintone のアプリやレコードに対して権 限がないなど、kintone の設定とカスタマイズに齟齬がないか確認する必要が あります。

このエラーの場合、[kintone からの応答] に、「入力内容が正しくありません。 Code:CB_VA01 record. 氏名 .value: 必須です。」とありますので、レコード を追加・更新する際に kintone アプリで必須と指定されているフィールドに 値をセットしていないためエラーが発生していることがわかります。

このように、エラーメッセージを読むとエラーの原因が判明する、または解決 の糸口が見つかることはよくあります。一度心を落ち着けて、エラーメッセー ジを読んでみてください。

また、<u>サポートサイト</u>では、カスタマインでカスタマイズしているとよく出会 うエラーメッセージとその原因を記事としてまとめています。 サポートサイト

(カスタマインの設定画面右上「サポートサイト」からも開くことができます) https://support.gusuku.io/

サポートサイト上部の検索窓で、エラーと検索してみてください。

	ISUKU SUPPOI コードでらくらくkintoneフ	't カスタマイズのCust	omineと、キントーンアプリの配布・バックアップ
ホーム	ナレッジベース	お知らせ	
Q エラ 検索キーワー キーワードの)	ラー ドを入力してEnterキーで検 選び方のコツなどは、こちら	索してください。 5の「効果的な検索だ	5法について」を参考にしてみてください。

検索すると、エラーに関する記事が表示されます。例えば、先ほどご紹介 した必須エラーの例は下記の「入力内容が正しくありません。records[0]. OO.value: 必須です」というエラーが出ました。 に詳しく記載していますの で、ぜひ確認してみてください。

 「クエリ記法が間違っています。」というエラーが出ました。 https://support.gusuku.io/ja-JP/support/solutions/articles/36000218833 画像のようなエラーが出たときの解決手順をご紹介します。解決手順1. 「クエリで条件を指定してレコードを取得する」を使っているアクションを探すこのエラーが出ているということは、「クエリで条件を指
 「入力内容が正しくありません。records[0].OO.value: 必須です」というエラーが出ました。 https://support.gusuku.io/ja-JP/support/solutions/articles/36000220365 こちらは、追加もしくは更新するレコードの中の入力必須のフィールドが空になる場合に発生するエラーです。解決手順1.エラーが出るアクションの 候補を探すまずはこのエラーが出そうなアクションを探します
 エラーを表示してレコードの保存を中断する方法 https://support.gusuku.io/ja-JP/support/solutions/articles/36000303352 Customineで提供しているエラーを表示するやることは、大きく分けて三つの種類があります。 フィールドにエラーを表示する、レコードにエラーをセットする、エラーダイアログを表示するの三種類です。

ある程度の長い文で検索して想定する記事がヒットしない場合は、検索キー ワードの単語をスペースで区切って指定いただくのがおすすめです。 例えば今回の必須エラーであれば、「必須 エラー」のように区切って検索し てみてください。自然な文章をキーワードとして検索するより目当ての記事が 検索されやすいことが多いです。

6-4. 原因究明に役立つカスタマインの便利な機能

カスタマインは、うまく動かないときの原因究明に役立つ機能を提供していま す。

アクショングラフ

アクショングラフは、<条件:他のアクションの実行が完了した時>などで順番を指定したアクションが、どの順番で実行されるのかを図で表示してくれる機能です。

図でアクションの関連性を見て確認できるので、アクションが想定通りにつな がっているかを確認したい場合に便利です。

カスタマイズを実行した時、順番に実行されるアクションが途中で動かない場 合は、アクショングラフを確認してみてください。矢印が途切れていたり、順 番が間違っているなど、一目でわかるためとても便利です。



アクショングラフ上に表示されている番号は、アクション番号です。なおアク ショングラフ上のアクション番号をクリックすると、該当するアクションの設 定にジャンプすることができます。 アクショングラフは、ページメニューの「アクショングラフ」を選択または「ア クショングラフボタン」をクリックすると表示されます。

> > レコード追加時自動採番	•••• + •• 有効
ページ一覧	7
- ページを追加	
ページをコピー	
ページを削除	
指定アクションに移動	
アクショングラフ	882

アクションの無効化

一時的にアクションが実行されないようにしたいときに使用します。 アクションを無効化した後で「kintone アプリへ登録」を行った際に、無効化 したアクションは kintone アプリに登録されず、実行されなくなります。

アクションの左上のアクション番号または右上の有効ボタンをクリックする と、そのアクションを無効化できます。

3	לגאלאב			「風」へ ・	••
	取得したレコードを絞り込む	i (0	他のアクションの実行が完了した時 🕜	
	元のレコード	2		アクション 2	
	条件になるフィールド	顧客ランク			
	条件	等しい		東件を追加	
	比較僅	A			

右上のメニューから 無効にする を選択する方法もあります。

他のアクションの実行が完了した時空のアクションを上に挿入アクション空のアクションを下に挿入アクションを複製アクションを複製条件を追加アクションを削除無効にする	他のアクションの実行が完了した時 空のアクションを上に挿入 アクション 空のアクションを下に挿入 アクション アクションを複製 条件を追加 第効にする			有効 ヘ ・・・	
アクション 空のアクションを下に挿入 アクション アクションを複製 条件を追加 アクションを削除 無効にする 1000000000000000000000000000000000000	アクション 空のアクションを下に挿入 アクション アクションを複製 条件を追加 アクションを削除 無効にする 無効にする		他のアクションの実行が完了した時	空のアクションを上に挿入	
アクション 2 アクションを複製 条件を追加 アクションを削除 無効にする	アクション 2 アクションを複製 条件を追加 アクションを削除 無効にする			空のアクションを下に挿入	
条件を追加 アクションを削除 無効にする	条件を追加 アクションを削除 無効にする		アクション 2	アクションを複製	
無効にする	無効にする		条件を追加	アクションを削除	
				無効にする	
セットチョンには、主くにのじフィールトコートをセット セット先のフィールト		挿入 ✔ = 請求先会	社名	▶ 請求先会社名 文字	四 (1行)

無効化したアクションは、左上のアクション番号がグレーになり、右上には無 効と表示されます。

3	コメント入力			100 ••••
E	取得したレコードを絞り込む	0	他のアクションの実行が完了した時	0
			2 22	

他のアクションの動きを確認するために無効化したり、アクションの設定を変 更して動きを確認する際に、変更前の設定を残しておきたい場合にも使用でき ます。

その際は、アクションを複製して、複製元のアクションを無効化して残してお く、という方法がおすすめです。

アクションの複製は下記のメニューから行ってください。

	有効 へ ・・・
他のマクションの実行が空フした時	空のアクションを上に挿入
	空のアクションを下に挿入
アクション 2	アクションを複製
条件を追加	アクンヨンを削除
	無効にする

ページの無効化

個別のアクションではなく、ページをまるごと無効化・有効化することができ ます。

無効化したページは、「kintone アプリへ登録」する際に kintone アプリに登録されず、実行されません。

ページの無効化は、カスタマイズ画面右上の 有効 をクリックすると無効に切 り替えできます。



または、ページの選択・並べ替え をクリックすると表示される ページ選択 画 面でも無効化できます。



このページ選択 画面では、複数ページをまとめて設定できます。ページが一 覧で表示されますので、どのページが有効・無効になっているか確認する場合 にも便利です。

ページ			
	テーブルに顧客アプリのレコードを追 加	00	無效
\bigcirc	顧客連絡アプリにレコード登録	00	有效
		ОК	キャンセル

ページメニューから無効化することもできます。

ページ ?		
< < 1/2 >	▶ テーブルに顧客アプリのレコード…	••• + 有効
	ページー覧	
	ページを追加	有効 へ ••••
追加画面	ページをコピー	?
一覧画面の	ページを削除	80
条件を追	指定アクションに移動	
	アクショングラフ	
	ページを画像として保存	有効 ヘ ・・・
ボタンを	ページを無効にする	2

ページの無効化は、他のページの動作を確認したいときに、一旦ページ全体を 無効化しておきたい場合などに使用します。この機能を効果的に使うためには、 機能ごと、画面ごとにこまめにページを分けてカスタマイズを作成しておくこ とが必要です。 またページの無効化は、次に記載するページのコピーと組み合わせて使用する ことも多い機能です。

ページのコピー

ページ内のカスタマイズに変更を加える場合は、ページをコピーして無効化し、 変更前のカスタマイズの状態を保存しておくことをおすすめします。 変更前の状態を保存しておけば、カスタマイズの変更に失敗した場合などに、 コピーしておいたページを有効化し、変更したページを無効化することでカス タマイズを変更前の状態に簡単に戻すことができます。



6-5. レコードが想定通り取得・絞り込みできているか

ポップアップ表示して確認する

[やること:<u>キーを指定してレコードを取得する</u>] や [やること:<u>クエリで条</u> <u>件を指定してレコードを取得する</u>] のように、条件を指定してレコードを取得 するやることを使用した場合や、取得したレコードを [やること:<u>取得したレ</u> <u>コードを絞り込む</u>] で絞り込んだ場合、想定通りのレコードが取得・絞り込み できているか確認したい事がよくあります。

そういった場合は、本書の例でも取り上げているように取得・絞り込んだ後の レコードを [やること:レコードの一覧をポップアップで表示する] で確認し てみてください。

3021306	
専得したレコードを絞り込む	 他のアクションの実行が完了した時 (?)
R0L3-F 7	799a> T
条件になるフィールド おびれ日	
条件 注注れる	条件を 400
定規値 中央区	
(SKSFAD	
レコードの一覧をポップアップで表示する	
6-3-F	79585 8
教育するフィールド ぜぶれん	teshte,
潮水方法 甲一进行	(3オボタン)
メッセージ入力 く入力さ	UREA)
0K59ンの名前 -0K	
キャンセルボタンの名前 キャンセ	
金融家を許可するたどうか ままらつ	
	#告诉问:# 告

このように、絞り込んだ結果を一覧で表示できるため、想定通りの絞り込みが 行われているかを一目で確認することが可能です。

	都道府県名	市区町村名	
\bigcirc	島根県	川本町	
0	和歌山県	串本町	
\bigcirc	奈良県	田原本町	
\bigcirc	大阪府	島本町	
\bigcirc	静岡県	川根本町	
C	OK キャンセル		

確認のための仮のアクションですので、確認が終わった時にはアクションを 無効にしておけば、仮にそのあと条件が変わった場合など、再確認の際にも役 立ちます。

6-6. テスト用アプリ作成

カスタマイズをガンガン試してと言われても、既にユーザーが使っている kintone アプリ (以下 本番用アプリ)にカスタマイズすると、誤ってレコード を更新してしまうリスクがあったり、フィールドの値やステータスなど、カス タマイズを試したい状態のレコードがなく実行確認が難しいことなどがよくあ ります。

そういった時のために、実際にユーザーが使うアプリとは別に、そのアプリを コピーしたテスト用アプリを作成することをおすすめします

