

gusuku Customine

カスタマイズの歩き方

Excel 関数とカスタマイズ編



なまえ

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（以降カスタマイズと表記）。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ズ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。

この冊子は、Excelの関数を駆使して実現していたことを、kintoneで解決できるか悩んでいる方や、Excelの関数や機能と照らしあわせながらカスタマイズを学習したい方に向けて制作しています。

なお本冊子では、イメージを掴みやすくするために、Excelのデータ形式をそのままkintoneアプリで再現していますが、実際のkintoneアプリ開発では必ずしもExcelのデータと同一の構造のアプリにすることが最適とは限らない点にご注意ください。

この冊子で理解できること

- 1, Excel関数や機能の、gusuku Customineでの実現方法

目次

1. シナリオ	P 0 5
2. ミッションⅠ	P 0 8
現在の管理内容を損なわず、『注文管理』アプリの操作性を改善せよ！	
2-1. 注文管理の入力を簡単に：VLOOKUP	
2-2. 日付計算を自動に：EDATE	
2-3. 返却遅延を分かりやすく：条件付き書式	
2-4. 一覧画面で集計：SUMIF	
3. ミッションⅡ	P 2 7
『取引先マスタ』アプリでの、入力ミスの再発を防げ！	
3-1. 取引先マスタの入力補助：自動採番	
3-2. 入力ミス防止：入力規則（編集不可）	
4. ミッションⅢ	P 3 3
『注文管理』アプリの集計結果を『取引先マスタ』アプリに反映せよ！	
4-1. 集計：COUNTIFS	
5. 今回使用したやることまとめ	P 3 9
6. 知っておいていただきたい、カスタマイズの基礎知識	P 4 0
7. うまく動かないときは？	P 5 0

1. シナリオ

次のようなシナリオで、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。Excel の関数や機能との対比をイメージしやすくするために、Excel で行っていた業務と、それを kintone アプリでカスタマイズを作成して実現したときの動きを並べて、ストーリー仕立てで説明していきます。

ストーリー

- ・志摩氏は、アロマディフューザーのレンタルサービス会社のレンタル管理を行う部署に所属している。
- ・レンタルの管理は Excel で行われてきたが、会社の方針で kintone を導入することになった。
- ・Excel から kintone への移行はすでに完了している。
- ・志摩氏は、Excel を使用していた当時から、より業務が円滑に進むよう業務改善を行ってきた。
- ・kintone で業務を開始してからも、基本機能を駆使して業務改善を推進してきているが、基本機能ではできないこともあると知り、限界を感じていた。
- ・gusuku Customine を使用することで今は諦めている機能を実現できるかもしれないと知り、使い始めたばかり。

以前使用していた Excel の構成

元々の管理では、『取引先マスタ』『サービスマスタ』『注文管理』シートを運用していた。

『取引先マスタ』シート

	A	B	C	D	E
1	取引先コード	取引先名	担当者名	所在地	返却遅延回数
2	CU-001	金都運総研	下山 達士	岐阜県岐阜市××××	0
3	CU-002	林田商会	森 惇	埼玉県浦和市××××	0
4	CU-003	板橋電子株式会社	末永 妃里	神奈川県小田原市××××	0
5	CU-004	新山物産	金子 真帆	大阪府大阪市北区梅田××××	1
6	CU-005	岩下税理士事務所	佐々木 樹里	兵庫県神戸市須磨区××××	1

- ・取引先の情報を登録するシートで、取引先には取引先コードを付与している。
- ・取引先ごとの返却遅延回数を表示している。

『サービスマスタ』シート

	A	B	C
1	サービス名	単価	月数
2	3ヶ月レンタル	¥10,000	3
3	6ヶ月レンタル	¥15,000	6
4	12ヶ月レンタル	¥25,000	12

- ・提供サービスとその単価などを登録している。

『注文管理』シート

	A	B	C	D	E	F	G	H	I	J	K	L
1	サービス名	売上金額合計										
2	3ヶ月レンタル	¥20,000										
3	6ヶ月レンタル	¥45,000										
4	12ヶ月レンタル	¥50,000										
5												
6	受注日	注文番号	取引先コード	取引先名	担当者名	サービス名	契約月数	単価	終了日	リマインド日	返却日	返却遅延
7	2022/2/24	CU-001-20220224	CU-001	金都運総研	大垣 加奈	3ヶ月レンタル	3	¥10,000	2022/5/24	2022/4/24	2022/05/24	
8	2022/3/3	CU-004-20220303	CU-004	新山物産	金子 真帆	6ヶ月レンタル	6	¥15,000	2022/9/3	2022/8/3	2022/10/01	返却遅延
9	2022/7/10	CU-003-20220710	CU-003	板橋電子株式会社	未永 妃里	6ヶ月レンタル	6	¥15,000	2023/1/10	2022/12/10	2023/01/10	
10	2023/1/20	CU-005-20230120	CU-005	岩下税理士事務所	佐々木 樹里	3ヶ月レンタル	3	¥10,000	2023/4/20	2023/3/20	2023/04/25	返却遅延
11	2023/3/13	CU-003-20230313	CU-003	板橋電子株式会社	未永 妃里	12ヶ月レンタル	12	¥25,000	2024/3/13	2024/2/13		
12	2023/5/24	CU-002-20230524	CU-002	林田商会	森 惇	6ヶ月レンタル	6	¥15,000	2023/11/24	2023/10/24		
13	2023/5/31	CU-001-20230531	CU-001	金都運総研	中谷 あおい	12ヶ月レンタル	12	¥25,000	2024/5/31	2024/4/30		

- ・ レンタルの注文内容や期間を管理している。
- ・ シート上部で、サービス名ごとの売上金額を表示している。
- ・ 返却遅延の場合は、条件付き書式で強調表示を行ってセルを目立たせている。

kintone アプリ化

Excel での運用を踏まえ、『取引先マスタ』『サービスマスタ』『注文管理』のアプリを作成した。

『取引先マスタ』アプリ

	取引先コード↑	取引先名	担当者名	所在地	返却遅延回数	
	CU-001	金都運総研	下山 達士	岐阜県多治見市××××		 
	CU-002	林田商会	森 惇	埼玉県浦和市××××		 
	CU-003	板橋電子株式会社	未永 妃里	神奈川県小田原市××××		 
	CU-004	新山物産	金子 真帆	大阪府大阪市北区梅田××××		 
	CU-005	岩下税理士事務所	佐々木 樹里	兵庫県神戸市須磨区××××		 

『サービスマスタ』アプリ

	サービス名	単価↑	
	3ヶ月レンタル	10,000	 
	6ヶ月レンタル	15,000	 
	12ヶ月レンタル	25,000	 

『注文管理』アプリ

	受注日↑	取引先コード	注文番号	取引先名	担当者名	サービス名	単価	終了日	リマインド日	返却日	返却遅延
	2022-02-24	CU-001	CU-001-20220224	金都理総研	大垣 加奈	3ヶ月レンタル	10,000	2022-05-24	2022-04-24	2022-05-24	
	2022-03-03	CU-004	CU-004-20220303	新山物産	金子 真帆	6ヶ月レンタル	15,000	2022-09-03	2022-08-03	2022-10-01	返却遅延
	2022-07-10	CU-003	CU-003-20220710	板橋電子株式会社	未永 妃里	6ヶ月レンタル	15,000	2023-01-10	2022-12-10	2023-01-10	
	2023-01-20	CU-005	CU-005-20230120	岩下税理士事務所	佐々木 樹里	3ヶ月レンタル	10,000	2023-04-20	2023-03-20	2023-04-25	返却遅延
	2023-03-13	CU-003	CU-003-20230313	板橋電子株式会社	未永 妃里	12ヶ月レンタル	25,000	2024-03-13	2024-02-13		
	2023-05-24	CU-002	CU-002-20230524	林田商会	森 惇	6ヶ月レンタル	15,000	2023-11-24	2023-10-24		
	2023-05-31	CU-001	CU-001-20230531	金都理総研	大垣 加奈	12ヶ月レンタル	25,000	2024-05-31	2024-04-30		

2. ミッションI 現在の管理内容を損なわず、 『注文管理』アプリの操作性を改善せよ!

2-1. 注文管理の入力を簡単に：VLOOKUP

志摩氏はまず、『注文管理』アプリの入力をスムーズに行えるようなカスタマイズを試してみることになりました。

これまでの『注文管理』シートの運用

<code>=VLOOKUP(C7[取引先コード], 取引先マスタ!A:C,2,FALSE)</code>			<code>=VLOOKUP(C7[取引先コード], 取引先マスタ!A:C,3,FALSE)</code>		
取引先コード	取引先名	担当者名	取引先マスタシート		
CU-001	金都運総研	大垣 加奈	A	B	C
CU-004	新山物産	金子 真帆	取引先コード	取引先名	担当者名
CU-003	板橋電子株式会社	未永 妃里	CU-001	金都運総研	下山 達士
CU-005	岩下税理士事務所	佐々木 樹里	CU-002	林田商会	森 惇
CU-003	板橋電子株式会社	未永 妃里	CU-003	板橋電子株式会社	未永 妃里
CU-002	林田商会	森 惇	CU-004	新山物産	金子 真帆
CU-001	金都運総研	中谷 あおい	CU-005	岩下税理士事務所	佐々木 樹里

運用:取引先コードをキーに、『取引先マスタ』から取引先名・担当者名を Excel の VLOOKUP 関数（以下、VLOOKUP と記載）で検索し、指定のセルに入力を行っている。

ルール①：会社名の変更がある場合は『取引先マスタ』シートを更新する。

ルール②：担当者名は注文により異なるケースが多くあるため、都度『取引先マスタ』シートの変更は行わず、『注文管理』シート上で上書きする。

ルール③：レンタルが終了した取引については VLOOKUP を削除して値を固定し、『取引先マスタ』シートに変更があっても上書きしない。

志摩氏のチャレンジ：基本機能のルックアップを活用してみる

『注文管理』アプリでは、VLOOKUP を用いていた箇所を kintone の基本機能のルックアップで設定しました。これにより取引先コードをもとに、『取引先マスタ』アプリから、取引先名と担当者名をコピーする動きが実装できました。

『注文管理』アプリ

取引先コード	取引先名	担当者名
CU-001	取得 クリア 金都運総研	下山 達士

課題 1：『取引先マスタ』アプリの変更が、 『注文管理』アプリに自動で反映されない

ところが、『取引先マスタ』アプリの値を変更しても VLOOKUP の動きとは異なり、『注文管理』アプリのレコードが更新されません。これは、kintone のルックアップが、その時点での情報をコピーをする機能であるためです。そのため、すでに作成したレコードに対して『取引先マスタ』アプリの変更を反映させたいときには、毎回ルックアップの取得ボタンを押して最新の情報に更新する運用にしていました。

『取引先マスタ』アプリ

取引先コード	CU-001	担当者名を「大垣 加奈」から 「下山 達士」に変更しても
取引先名	金都運総研	
		担当者名
		下山 達士

『注文管理』アプリ

取引先コード	取引先名	担当者名
CU-001	金都運総研	大垣 加奈

担当者名の変更が
自動的に反映されない

しかし、毎回手動で更新することは面倒ですし、取得漏れも発生していました。

これを改善するため、志摩氏はまず『取引先マスタ』アプリにおける変更に伴い、『注文管理』アプリのレコードも自動で変更されて欲しいと考え、カスタマイズのサポートサイトで記事を検索してみました。

すると、「[ルックアップしているフィールドの自動更新をする kintone のカスタマイズ](#)」という記事を見つけ、参考にしながらカスタマイズをしてみました。

カスタマイズしてみた

カスタマイズするアプリ：『取引先マスタ』

13	
他アプリのルックアップを取得しなおす	レコードを保存した直後（削除後は除く）
アプリ	[Excel慣れ]注文管理
ルックアップフィールド	取引先コード
ルックアップの値	=取引先コード

実行結果

『注文管理』アプリ

	受注日↑	取引先コード	注文番号	取引先名	担当者名
	2022-02-24	CU-001	CU-001-20220224	金都運総研	大垣 加奈
	2022-03-03	CU-004	CU-004-20220303	新山物産	金子 真帆
	2022-07-10	CU-003	CU-003-20220710	板橋電子株式会社	未永 妃里
	2023-01-20	CU-005	CU-005-20230120	岩下税理士事務所	佐々木 樹里
	2023-03-13	CU-003	CU-003-20230313	板橋電子株式会社	未永 妃里
	2023-05-24	CU-002	CU-002-20230524	林田商会	森 惇
	2023-05-31	CU-001	CU-001-20230531	金都運総研	大垣 加奈



『取引先マスタ』アプリ

取引先コード	CU-001	担当者名を「大垣 加奈」から「下山 達士」に変更すれば
取引先名	金都運総研	担当者名 下山 達士



『注文管理』アプリ

	受注日↑	取引先コード	注文番号	取引先名	担当者名
	2022-02-24	CU-001	CU-001-20220224	金都運総研	下山 達士
	2022-03-03	CU-004	CU-004-20220303	新山物産	金子 真帆
	2022-07-10	CU-003	CU-003-20220710	板橋電子株式会社	未永 妃里
	2023-01-20	CU-005	CU-005-20230120	岩下税理士事務所	佐々木 樹里
	2023-03-13	CU-003	CU-003-20230313	板橋電子株式会社	未永 妃里
	2023-05-24	CU-002	CU-002-20230524	林田商会	森 惇
	2023-05-31	CU-001	CU-001-20230531	金都運総研	下山 達士

これで課題1はクリアできました。

しかしこのカスタマイズだと、すでにレンタルが終了している（商品が返却済）注文も情報が更新されてしまうため、このままではルール③（レンタルが終了した取引については値を固定し、『取引先マスタ』シートに変更があっても上書きしない。）に対応できていません。

課題2：『取引先マスタ』アプリの変更が、 『注文管理』アプリの全てのレコードに自動で反映されてしまう

志摩氏は更に、これまでのカスタマイズに加えてルール③に対応させるため、カスタマイズの修正を行いました。

カスタマイズしてみた

カスタマイズするアプリ：『取引先マスタ』

38

条件を組み立ててレコードを取得する

取得元アプリ [Excel慣れ]注文管理

検索条件 (返却日 = "")
and (取引先コード = "\${取引先コード}")

最大取得件数 (入力されていません)

レコードを保存した直後 (削除後は除く)

条件1 返却日 ↓ = (等しい) ↓ 日付を指定 ↓ YYYY-MM-DD

条件2 取引先コード ↓ = (等しい) ↓ \${取引先コード}

39

レコードを書き出す

レコード 38

書き出し先アプリ [Excel慣れ]注文管理

マッピング 取引先コード=取引先コード

更新または追加 既存レコードを更新

他のアクションの実行が完了した時

アクション 38

実行結果

『注文管理』アプリ

受注日↑	取引先コード	注文番号	取引先名	担当者名	サービス名	単価	終了日	リマインド日	返却日
2022-02-24	CU-001	CU-001-20220224	金都運総研	大垣 加奈	3ヶ月レンタル	10,000	2022-05-24	2022-04-24	2022-05-24
2022-03-03	CU-004	CU-004-20220303	新山物産	金子 真帆	6ヶ月レンタル	15,000	2022-09-03	2022-08-03	2022-10-01
2022-07-10	CU-003	CU-003-20220710	板橋電子株式会社	未永 妃里	6ヶ月レンタル	15,000	2023-01-10	2022-12-10	2023-01-10
2023-01-20	CU-005	CU-005-20230120	岩下税理士事務所	佐々木 樹里	3ヶ月レンタル	10,000	2023-04-20	2023-03-20	2023-04-25
2023-03-13	CU-003	CU-003-20230313	板橋電子株式会社	未永 妃里	12ヶ月レンタル	25,000	2024-03-13	2024-02-13	
2023-05-24	CU-002	CU-002-20230524	林田商会	森 惇	6ヶ月レンタル	15,000	2023-11-24	2023-10-24	
2023-05-31	CU-001	CU-001-20230531	金都運総研	下山 達士	1		2024-05-31	2024-04-30	

これで、レンタルが終了されている注文のみ、『取引先マスタ』アプリで変更があった場合、その内容に更新する動きを作ることができましたが、ここで新たな課題が出てきました。

課題3：『注文管理』アプリの担当者名フィールドが編集できない

ルール②（担当者名は注文により異なるケースが多くあるため、都度『取引先マスタ』シートの変更は行わず、『注文管理』シート上で上書きする。）に対応するためには、これまで通り『注文管理』アプリ上で担当者名を変更できるようにする必要があります。

しかし、ルックアップコピー先のフィールドは、ルックアップの仕様で編集することができません。

『注文管理』アプリ

取引先コード	取引先名	担当者名
CU-001	取得 クリア 金都運総研	下山 達士

この課題についてもカスタミンで簡単に解決が図れそうだったので、志摩氏はカスタマイズを行いました。

カスタマイズしてみた

カスタマイズするアプリ：『注文管理』

7

フィールドを有効化する	編集画面を表示した時
フィールド 担当者名	一覧画面の鉛筆を押した時は含めない

実行結果

『注文管理』アプリ

取引先コード	取引先名	担当者名
CU-001	取得 クリア 金都運総研	下山 達士

これで、担当者名を自在に『注文管理』アプリ上で変更することができるようになり、業務改善が順調に進んできたと思いきや、ここで想定外の問題が発生しました。

課題4：『注文管理』アプリ上で変更した担当者名が、『取引先マスタ』アプリの情報に上書きされてしまう

というのも、担当者名を『注文管理』アプリで確かに変更したはずなのに、担当者名が『取引先マスタ』アプリの担当者名に戻ってしまっているという連絡が他の担当者から来ました。

志摩氏が kintone の変更履歴を調べたところ、どうやら『取引先マスタ』アプリの所在地フィールドを編集したタイミングで、『注文管理』アプリの担当者名が『取引先マスタ』アプリ側の情報に上書きされているようです。

さらに調査を進めると、ルックアップの自動更新のカスタマイズにより、『取引先マスタ』アプリでレコードが更新された際、『注文管理』アプリの担当者名フィールドが『取引先マスタ』アプリの情報に自動更新されていたことが原因だとわかりました。

実務担当者と議論した結果、業務要件としては自動更新も、担当者名を可変にすることもそれぞれ必要だったので、『注文管理』アプリで担当者名を変更したレコードについてはそれを識別できるようなフラグを立てるようにし、『取引先マスタ』アプリの情報が変更されても、自動更新の対象外とすることになりました。

これを踏まえて手入力したレコードであることを識別するために、『注文管理』アプリに新たにチェックボックスフィールドを追加したうえ、カスタマイズを変更しました。

『注文管理』アプリ

取引先コード	取引先名	担当者名	自動更新対象外
CU-001	金都連総研	下山 達士	<input type="checkbox"/> 自動更新対象外

カスタマイズしてみた

カスタマイズするアプリ：『取引先マスタ』

38

条件を組み立ててレコードを取得する

取得元アプリ [Excel慣れ]注文管理

検索条件
(返却日 = "")
and (取引先コード = "\${取引先コード}")
and (自動更新対象外 not in ("自動更新対象外"))

最大取得件数 (入力されていません)

レコードを保存した直後 (削除後は除く)

条件1 返却日 = (等しい) 日付を指定 YYYY-MM-DD 挿入

条件2 取引先コード = (等しい) \${取引先コード} 挿入

条件3 自動更新対象外 次のいずれも含まない 自動更新対象外 挿入

39

レコードを書き出す

レコード 38

書き出し先アプリ [Excel慣れ]注文管理

マッピング 取引先コード=取引先コード

更新または追加 既存レコードを更新

他のアクションの実行が完了した時

アクション 38

さらに、自動更新対象外にチェックがついた場合のみ、担当者名フィールドを有効化するカスタマイズを追加しました。

カスタマイズするアプリ：『注文管理』

43

フィールドを有効化する
フィールド 担当者名

編集画面を表示した時
一覧画面の鉛筆を押した時 一覧画面の鉛筆を押した時は含まない

フィールド値が特定の値ならば
フィールド 自動更新対象外
条件 等しい
比較値 自動更新対象外

44

フィールドを有効化する
フィールド 担当者名

フィールドの値を編集して値が変わった時
フィールド 自動更新対象外
一覧画面の鉛筆を押した時 一覧画面の鉛筆を押した時は含まない

フィールド値が特定の値ならば
フィールド 自動更新対象外
条件 等しい
比較値 自動更新対象外

46

フィールドを無効化する
フィールド 担当者名

フィールドの値を編集して値が変わった時
フィールド 自動更新対象外
一覧画面の鉛筆を押した時 一覧画面の鉛筆を押した時は含まない

フィールド値が特定の値でないならば
フィールド 自動更新対象外
条件 等しい
比較値 自動更新対象外

実行結果

『注文管理』アプリ

取引先コード	取引先名	担当者名	自動更新対象外
CU-001	金都運総研	中谷 あおい	<input checked="" type="checkbox"/> 自動更新対象外

自動更新対象外に
チェックを入れておく

『取引先マスタ』アプリ

取引先コード	CU-001
取引先名	金都運総研
担当者名	下山 達士
所在地	岐阜県岐阜市xxxx

所在地変更

取引先コード	CU-001
取引先名	金都運総研
担当者名	下山 達士
所在地	岐阜県多治見市xxxx

『注文管理』アプリ

取引先コード	CU-001
取引先名	金都運総研
担当者名	中谷 あおい
自動更新対象外	自動更新対象外

『取引先マスタ』が更新されても、ルックアップが自動更新されず、担当者名が上書きされない。

まとめ

kintone の基本機能のルックアップで、VLOOKUP と同様にマスタ側の情報を参照先アプリにコピーする動きは実現できます。

ただし VLOOKUP では、マスタ側の情報を変更すると参照先のデータも自動的に変更されますが、kintone のルックアップは、マスタ側を更新してもコピー先の情報は自動的に更新されません。これは、kintone のルックアップが、あくまでルックアップの取得ボタンを押した時点の情報を、参照先アプリにコピーするという仕組みになっているため、再度取得ボタンを押さない限り、最新の値に上書きされることはありません。

カスタマイズでは、[やること : [他アプリのルックアップを取得しなおす](#)] を用いることで、マスタ側アプリの情報変更に伴い、コピー先アプリの情報も自動で更新できるようになります。(課題 1 : 『取引先マスタ』アプリの変更が、『注文管理』アプリに自動で反映されない)

その一方、運用上の理由で特定条件の場合のみ、自動更新をしたいという要件も想定されます。その場合は、その条件に該当するレコードを [やること : [条件を組み立ててレコードを取得する](#)] で取得し、[やること : [レコードを書き出す](#)] で、条件に合致するレコードのみ最新の情報に更新するというカスタマイズを行えば実現することができます。(課題 2 : 『取引先マスタ』アプリの変更が、『注文管理』アプリの全てのレコードに自動で反映されてしまう)

さらに、ルックアップでコピーするフィールドは、kintone の仕様により編集ができないよう無効化されます。ですが、必ずしもマスタの情報を書き換えずに、そのときだけ情報を変更したいという要件もあるかと思えます。その場合は [やること : [フィールドを有効化する](#)] で、任意のフィールドを編集可能な状態にすることができます。(課題 3 : 『注文管理』アプリの担当者名フィールドが編集できない)

注意点として、個別にフィールドを編集することができるようになりましたが、マスタアプリでレコードの編集を行い、この時に自動更新のカスタマイズが動いた場合は、マスタ側の情報に上書きされてしまいます。

今回のシナリオでは、自動更新とフィールドの有効化のいずれの要件も残す必要性があり、折衷案として自動更新する対象のレコードかどうかを識別するためのチェックボックスをもうけ、チェックがついているレコードのみ自動更新するカスタマイズに変更することにしました。(課題4:『注文管理』アプリ上で変更した担当者名が、『取引先マスタ』アプリの情報に上書きされてしまう)

おさらい：VLOOKUP 関数

関数の構文

=VLOOKUP (検索対象、検索する場所、返される値を含む範囲内の列番号、1/TRUE または 0/FALSE として示される近似または完全一致 - を返します)。

関数の説明

配列の左端列で特定の値を検索し、対応するセルの値を返します。

引用

<https://support.microsoft.com/ja-jp/office/vlookup-%E9%96%A2%E6%95%B0-0bbc8083-26fe-4963-8ab8-93a18ad188a1>

2-2. 日付計算を自動に：EDATE

続いて志摩氏は、レンタルの終了日や、そのリマインド日も、これまでと同様に自動で日付が入るようにしたいと考えました。

これまでの『注文管理』シートの運用

受注日	注文番号	取引先コード	取引先名	担当者名	サービス名	契約月数	単価	終了日	リマインド日	返却日	返却遅延
2022/2/24	CU-001-20220224	CU-001	金都運総研	大垣 加奈	3ヶ月レンタル	3	¥10,000	2022/5/24	2022/4/24	2022/05/24	
2022/3/3	CU-004-20220303	CU-004	新山物産	金子 真帆	6ヶ月レンタル	6	¥15,000	2022/9/3	2022/8/3	2022/10/01	返却遅延
2022/7/10	CU-003-20220710	CU-003	板橋電子株式会社	未永 妃里	6ヶ月レンタル	6	¥15,000	2023/1/10	2022/12/10	2023/01/10	
2023/1/20	CU-005-20230120	CU-005	岩下税理士事務所	佐々木 樹里	3ヶ月レンタル	3	¥10,000	2023/4/20	2023/3/20	2023/04/25	返却遅延
2023/3/13	CU-003-20230313	CU-003	板橋電子株式会社	未永 妃里	12ヶ月レンタル	12	¥25,000	2024/3/13	2024/2/13		
2023/5/24	CU-002-20230524	CU-002	林田商会	森 惇	6ヶ月レンタル	6	¥15,000	2023/11/24	2023/10/24		
2023/5/31	CU-001-20230531	CU-001	金都運総研	中谷 あおい	12ヶ月レンタル	12	¥25,000	2024/5/31	2024/4/30		

=EDATE(A7[受注日],G7[契約月数])

=EDATE(I7[終了日],-1)

運用：入力された日付を元に Excel の EDATE 関数（以下、EDATE と記載）で日付計算し設定している

ルール①：受注日に EDATE で契約月数を足し、終了日を自動計算する。

ルール②：終了日の 1 か月前を EDATE で自動計算し、リマインド日に設定する。

志摩氏のチャレンジ：計算式フィールドで自動計算

「受注日」フィールドをもとに、計算フィールドで計算式を用いて、まずは「終了日」の設定を試みました。

『注文管理』アプリで終了日を計算

計算の設定
×

フィールド名 *

終了日

フィールド名を表示しない

計算式 * 計算式の書き方 利用できる関数

受注日 + 60*60*24*契約月数*30

計算式を表示しない

数値（例：1000）

数値（例：1,000）

日時（例：2012-08-06 2:03）

日付（例：2012-08-06）

時刻（例：2:03）

課題：計算式が複雑になってしまう

kintone での日付の計算式は秒単位で指定する必要があるため、一見どのような計算をしているのかが分かりづらくなってしまいう上に、1ヶ月は必ずしも30日ではないため、この計算式ではこれまでEDATEで算出していた日付と誤差が生じてしまいます。

例：受注日が2023年5月24日の場合

- ・上記計算式フィールドで計算した6ヶ月後の日付：2023年11月20日
- ・EDATEで計算した6ヶ月後の日付：2023年11月24日

もっと簡単に、それこそEDATEで設定するような感覚で日付の設定がしたい。志摩氏はカスタマイズを行ってみました。

カスタマイズしてみた

カスタマイズするアプリ：『注文管理』

10

日付を計算する

基準日 = 受注日

右辺値 = 契約月数

計算内容 ○か月後

セット先フィールド (省略可) 終了日

ルックアップで取得が成功した時

ルックアップフィールド サービス名

11

日付を計算する

基準日 = 終了日

右辺値 -1

計算内容 ○か月後

セット先フィールド (省略可) リマインド日

ルックアップで取得が成功した時

ルックアップフィールド サービス名

実行結果

『注文管理』アプリ

受注日 * 注文番号

2023-07-10 CU-005-20230710

取引先コード 取引先名 担当者名 自動更新対象外

CU-005 取得 クリア 岩下税理士事務所 佐々木 樹里 自動更新対象外

参照先からデータが取得されました。

サービス名 契約月数 単価

3ヶ月レンタル 取得 クリア 3 10000

参照先からデータが取得されました。

サービス名のルックアップの取得が成功すると、受注日を基準に終了日とリマインド日が自動計算され、それぞれのフィールドに結果がセットされる。

終了日 リマインド日 返却日 返却遅延

2023-10-10 2023-09-10 返却遅延

まとめ

kintone の基本機能でも、日付フィールドをもとに、計算フィールドで計算式を用いて「〇日後」といった日付の計算を行うことは可能ですが、日付フィールドの値は UNIX 時刻（1970 年 1 月 1 日 午前 0 時 0 分 0 秒からの経過秒数）として扱われるため、計算式も秒単位で指定する必要があります。例えば 30 日後の日付を入力したい場合、「日付 +60*60*24*30」と入力する必要があります。（課題：計算式が複雑になってしまう）

カスタミンの [やること : [日付を計算する](#)] を用いれば、右辺値（月）と計算内容（〇か月後なのか〇年後なのか等）を設定するだけで、自動で月の日数を考慮した日数計算を実現でき、EDATE と同じ要領で日付が計算できます。

Excel 関数とカスタミンの設定方法の対応表

Excel 関数 (EDATE)	カスタミン (やること)
開始日	日付を計算する パラメーター：基準日
月	日付を計算する パラメーター：右辺値, 計算内容

参考：1 か月後（30 日後）の日付を表示したい

https://jp.cybozu.help/k/ja/utility/app/calculate_date.html

おさらい：EDATE 関数

関数の構文

=EDATE(開始日, 月)

関数の説明

開始日から起算して、指定された月数だけ前または後の日付に対応するシリアル値を返します。

引用

<https://support.microsoft.com/ja-jp/office/edate-%E9%96%A2%E6%95%B0-3c920eb2-6e66-44e7-a1f5-753ae47ee4f5>

2-3. 返却遅延を分かりやすく：条件付き書式

さらに志摩氏は、返却遅延に対しては一目でわかるよう、これまで通り強調表示（セルの背景色変更、文字色変更、太字）をしたいと考えました。

もし返却遅延であれば強調表示を行う……IF 関数が頭をよぎりました。

これまでの『注文管理』シートの運用

終了日	リマインド日	返却日	返却遅延
2022/5/24	2022/4/24	2022/05/24	
2022/9/3	2022/8/3	2022/10/01	返却遅延
2023/1/10	2022/12/10	2023/01/10	
2023/4/20	2023/3/20	2023/04/25	返却遅延
2024/3/13	2024/2/13		
2023/11/24	2023/10/24		
2024/5/31	2024/4/30		



運用：返却遅延を Excel の条件付き書式で強調

志摩氏のチャレンジ：一覧画面で「返却遅延」フィールドを強調する

『注文管理』アプリ

受注日↑	取引先コード	注文番号	取引先名	担当者名	サービス名	単価	終了日	リマインド日	返却日	返却遅延
2022-02-24	CU-001	CU-001-20220224	金都運総研	下山 達士	3ヶ月レンタル	10,000	2022-05-24	2022-04-24	2022-05-24	
2022-03-03	CU-004	CU-004-20220303	新山物産	金子 真帆	6ヶ月レンタル	15,000	2022-09-03	2022-08-03	2022-10-01	返却遅延
2022-07-10	CU-003	CU-003-20220710	板橋電子株式会社	末永 妃里	6ヶ月レンタル	15,000	2023-01-10	2022-12-10	2023-01-10	
2023-01-20	CU-005	CU-005-20230120	岩下税理士事務所	佐々木 樹里	3ヶ月レンタル	10,000	2023-04-20	2023-03-20	2023-04-25	返却遅延
2023-03-13	CU-003	CU-003-20230313	板橋電子株式会社	末永 妃里	12ヶ月レンタル	25,000	2024-03-13	2024-02-13		
2023-05-24	CU-002	CU-002-20230524	林田商会	森 惇	6ヶ月レンタル	15,000	2023-11-24	2023-10-24		
2023-05-31	CU-001	CU-001-20230531	金都運総研	大垣 加奈	12ヶ月レンタル	25,000	2024-05-31	2024-04-30		

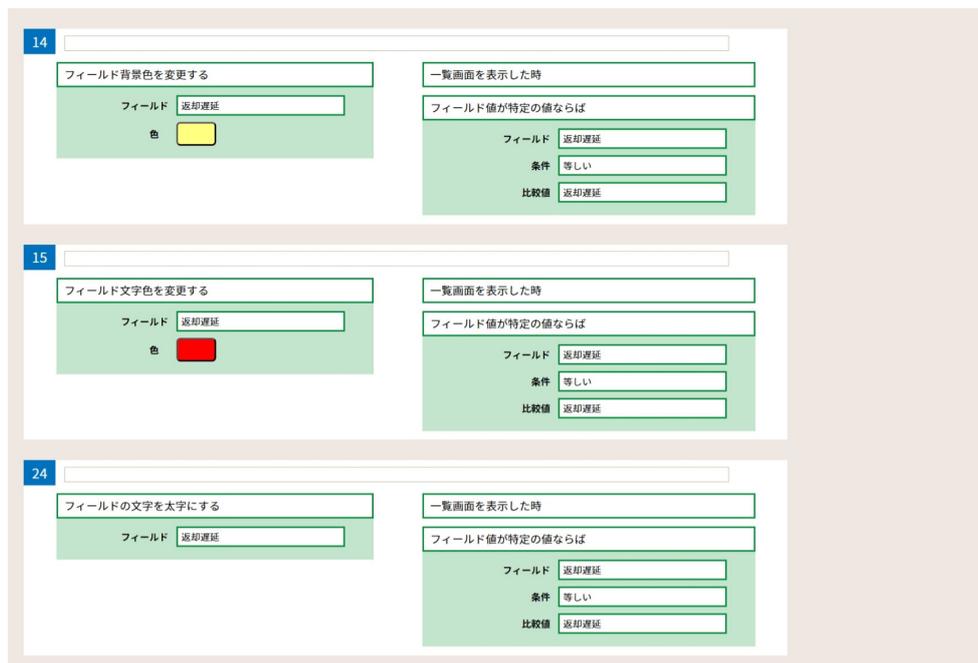
課題：基本機能ではフィールドの書式変更ができない

遅延返却フィールドを用意はしましたが、kintone の基本機能では Excel の条件付き書式設定のようなことはできないようです。このままだと、これまで使用してきた Excel に比べて視認性が劣ってしまいます。

何かいい方法があるはずだ、とカスタマイズのドキュメントページの「やること」一覧を見ているときに、「その他の UI」という項目があることに志摩氏は気が付きました。その中のやることを見ていくと、フィールドの背景色などを変更できるアクションがあると分かり、早速設定を試みました。

カスタマイズしてみた

カスタマイズするアプリ：『注文管理』



実行結果

『注文管理』アプリ

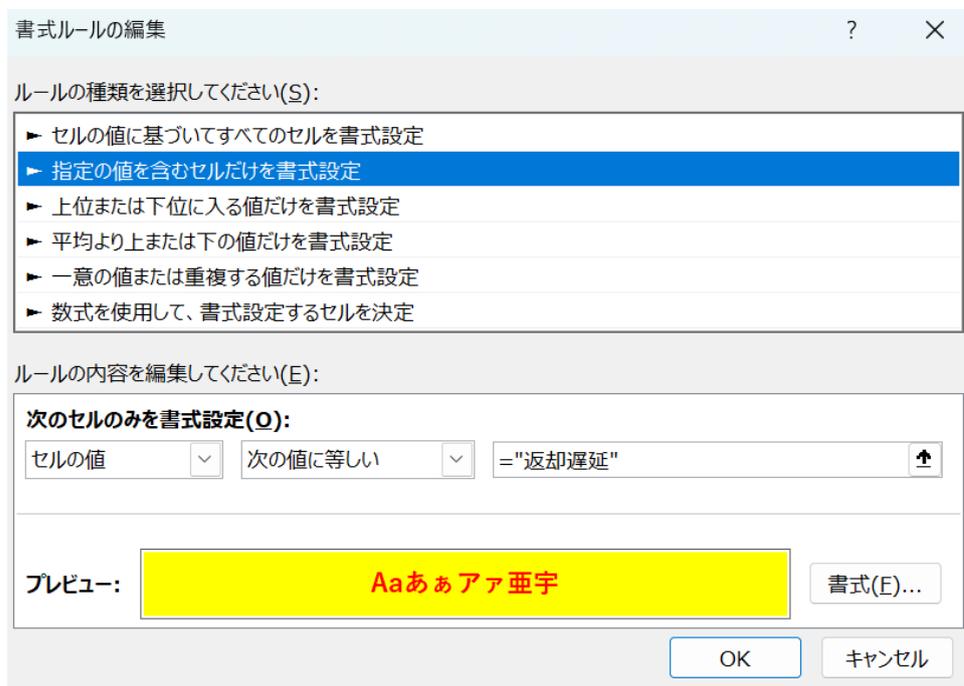
終了日	リマインド日	返却日	返却遅延	
2022-05-24	2022-04-24	2022-05-24		 
2022-09-03	2022-08-03	2022-10-01	返却遅延	 
2023-01-10	2022-12-10	2023-01-10		 
2023-04-20	2023-03-20	2023-04-25	返却遅延	 
2024-03-13	2024-02-13			 
2023-11-24	2023-10-24			 
2024-05-31	2024-04-30			 

まとめ

kintone の基本機能では実現できないような、文字色や背景色を変える、太字にするといったことはもちろん、カスタマイズを用いれば文字サイズを変更することもできます。(課題:基本機能ではフィールドの書式変更ができない)

また、この条件に合致する場合にこういった動作にしたい、という場合、[条件: [フィールド値が特定の値ならば](#)]を追加条件に用いることで、条件を絞り込むことができます。追加条件を増やすことで、さらに条件を絞り込むことも可能です。Excel の書式ルール編集画面で設定するルールの内容を、カスタマイズでは追加条件で設定します。

Excel の書式ルールの編集画面



カスタマイズの追加条件

フィールド値が特定の値ならば	
フィールド	返却遅延
条件	等しい
比較値	返却遅延

また Excel の条件付き書式でも IF 関数を用いて複雑な書式を設定することができますが、条件によっては関数が入れ子になり解読が困難になるケースがあります。カスタマイズではカスタマイズを日本語で設定するため、どのような条件設定がされているか分かりやすいというメリットがあります。

2-4. 一覧画面で集計：SUMIF

続いて志摩氏は、『注文管理』アプリ上での売上金額の集計に着手しました。

これまでの『注文管理』シートの運用

サービス名ごとに売上金額合計（単価）を集計

=SUMIF(\$F\$7:\$F\$13[サービス名], A2,\$H\$7:\$H\$13[単価])

A		
1	サービス名	売上金額合計
2	3ヶ月レンタル	¥20,000
3	6ヶ月レンタル	¥45,000
4	12ヶ月レンタル	¥50,000

F	G	H
サービス名	契約月数	単価
3ヶ月レンタル	3	¥10,000
6ヶ月レンタル	6	¥15,000
6ヶ月レンタル	6	¥15,000
3ヶ月レンタル	3	¥10,000
12ヶ月レンタル	12	¥25,000
6ヶ月レンタル	6	¥15,000
12ヶ月レンタル	12	¥25,000

運用：サービス名ごとに、Excel の SUMIF 関数（以下、SUMIF と記載）で売上金額合計（単価）を集計し、シート上部に表示している。

志摩氏のチャレンジ：一覧画面のレコード上部に文字や計算結果を表示する

『注文管理』シートは、皆が慣れ親しんできたシートなので、『注文管理』アプリでも同じように売上金額を確認できるようにすべく検討しました。表示位置も、なるべく変えたくはありません。

課題：基本機能ではレコード上部に文字を表示できない

ところが、kintone の基本機能では、レコードの上部に文字を表示することができないようです。

『注文管理』アプリ

受注日↑	取引先コード	注文番号	取引先名	担当者名	サービス名
2022-02-24	CU-001	CU-001-20220224	金都運総研	大垣 加奈	3ヶ月レンタル

こんな場所に文字を表示することが果たしてできるのだろうか、と志摩氏は一瞬思いましたが、以前見たドキュメントページで「表示する」関連のやることが複数あることを思い出しました。その中の [やること： [メニュー位置に文字を表示する](#)] を使えば実現できそうだと考え、カスタマイズに取り組みました。

カスタマイズしてみた

カスタマイズするアプリ：『注文管理』

1	<input type="text"/>	一覧の条件でレコードを全件取得する	一覧画面を表示した時
2	<input type="text"/>	レコード内の条件に合う行のフィールド合計値を計算する レコード選択アクション 1 計算するフィールド 単価 条件判定フィールド サービス名 条件 等しい 比較値 3ヶ月レンタル 結果をセットするフィールド (省略可) <選択されていません> 条件に合うレコードがなかった場合 ゼロをセット	他のアクションの実行が完了した時 アクション 1
34	<input type="text"/>	数値をフォーマットする 数値 =\$2 桁区切りを表示する 桁区切りを表示する 整数部の桁数 <入力されていません> 整数部をゼロ埋めするか 整数部をゼロ埋めする 小数点以下の桁数 <入力されていません> 小数部をゼロ埋めするか 小数部をゼロ埋めする セット先フィールド (省略可) <選択されていません>	他のアクションの実行が完了した時 アクション 2
3	<input type="text"/>	レコード内の条件に合う行のフィールド合計値を計算する レコード選択アクション 1 計算するフィールド 単価 条件判定フィールド サービス名 条件 等しい 比較値 6ヶ月レンタル 結果をセットするフィールド (省略可) <選択されていません> 条件に合うレコードがなかった場合 ゼロをセット	他のアクションの実行が完了した時 アクション 34
35	<input type="text"/>	数値をフォーマットする 数値 =\$3 桁区切りを表示する 桁区切りを表示する 整数部の桁数 <入力されていません> 整数部をゼロ埋めするか 整数部をゼロ埋めする 小数点以下の桁数 <入力されていません> 小数部をゼロ埋めするか 小数部をゼロ埋めする セット先フィールド (省略可) <選択されていません>	他のアクションの実行が完了した時 アクション 3
22	<input type="text"/>	レコード内の条件に合う行のフィールド合計値を計算する レコード選択アクション 1 計算するフィールド 単価 条件判定フィールド サービス名 条件 等しい 比較値 12ヶ月レンタル 結果をセットするフィールド (省略可) <選択されていません> 条件に合うレコードがなかった場合 ゼロをセット	他のアクションの実行が完了した時 アクション 35



実行結果

『注文管理』アプリ



サービス名：売上金額合計
 3ヶ月レンタル：¥ 20,000
 6ヶ月レンタル：¥ 45,000
 12ヶ月レンタル：¥ 50,000

	受注日 ↑	取引先コード	注文番号	取引先名	担当者名
	2022-02-24	CU-001	CU-001-20220224	金都運総研	大垣 加奈

まとめ

まず、[やること：[一覧の条件でレコードを全件取得する](#)]で、SUMIF の範囲指定にあたる、集計対象レコードの指定を行います。そして、[やること：[レコード内の条件に合う行のフィールド合計値を計算する](#)]で、検索条件および合計範囲を指定し、計算を行います。

また、kintone の基本機能では、一覧画面に文字を表示させることはできませんが、[やること：[メニュー位置に文字を表示する](#)]で、任意の文字列はもちろん、先行のアクションで計算した結果を表示させることができます。（課題：基本機能ではレコード上部に文字を表示できない）

なお、数値フィールドで数値を表示する場合は、kintone のフィールドの設定で桁区切りを設定することができますが、今回のようにメニュー位置に数値を表示させる場合には桁区切りの設定ができません。

せん。そのため [やること : [数値をフォーマットする](#)] カスタマイズを挟んで桁区切りを表示する設定を行い、その結果をメニュー位置に表示させています。

Excel 関数とカスタマイズの設定方法の対応表

Excel 関数 (EDATE)	カスタマイズ (やること)
範囲	一覧の条件でレコードを全件取得する
検索条件	レコード内の条件に合う行のフィールド合計値を計算する パラメーター：条件判定フィールド
合計範囲	レコード内の条件に合う行のフィールド合計値を計算する パラメーター：計算するフィールド

おさらい：SUMIF 関数

関数の構文

=SUMIF(範囲 , 検索条件 , [合計範囲])

関数の説明

指定した条件を満たす範囲内の値を合計

引用

<https://support.microsoft.com/ja-jp/office/sumif-%E9%96%A2%E6%95%B0-169b8c99-c05c-4483-a712-1697a653039b>

3. ミッションII『取引先マスタ』アプリでの、 入力ミスの再発を防げ!

3-1. 取引先マスタの入力補助：自動採番

『注文管理』アプリのカスタマイズにより、kintone での入力をスムーズに行えるようになりました。志摩氏は kintone とカスタマイズを組み合わせれば、Excel で諦めていたこともできるようになるかもしれないと考え、次はこれまで Excel で業務を行っていたなかで頻発していたトラブルの解消を目指しました。

これまでの『取引先マスタ』シートの運用

取引先コード	取引先名
CU-001	金都運総研
CU-002	林田商会
CU-003	板橋電子株式会社
CU-004	新山物産
CU-005	岩下税理士事務所

運用：取引先ごとに、取引先コードを手入力またはオートフィルで採番していた。

トラブル内容：取引先コードの抜けや重複。同一の取引先で複数の取引先コードを採番してしまう。

志摩氏のチャレンジ：取引先コードを自動で採番する

志摩氏はまず、レコード番号を使って自動採番が行えないかを検討してみました。レコード番号はレコードを作成したら自動的に付与されるため、これが使えると考えたのです。単位記号も設定できるようなので、うまくいきそうです。

『取引先マスタ』アプリ

計算の設定

? ヘルプ ? 設定手順の詳細 便利に使おうガイドブック vol.02 計算編

フィールド名 *

取引先コード

フィールド名を表示しない

計算式 * 計算式の書き方 利用できる関数

レコード番号

単位記号

CU- 前に付ける (例: ¥100) 後ろに付ける (例: 100km)

課題：レコード番号は計算フィールドの計算式に使用できない

ところが、設定を保存しようとするときエラーがでました。レコード番号は計算フィールドの計算式で使用できないためです。

基本機能では自動採番の要件を叶えることができなさそうだと考えた志摩氏は、ここでもカスタマイズを試みました。

計算の設定

? ヘルプ ? 設定手順の詳細 便利に使おうガイドブック vol.02 計算編

• 計算式で利用できないフィールドタイプのフィールド(レコード番号)が指定されています。以下の項目は計算式で利用できません：レコード番号、ラベル、リッチエディター、文字列（複数行）、添付ファイル、リンク、ユーザー選択、組織選択、グループ選択、関連レコード一覧、作業中、ステータス、カテゴリ。

フィールド名 *

取引先コード

カスタマイズしてみた

カスタマイズするアプリ：『取引先マスタ』

1

自動採番を行う

レコードを保存する直前（削除時は除く）

フィールド	取引先コード
アプリ単位の採番	アプリ単位で採番
ゼロ埋め	する
桁数	3
前に付ける文字列（プレフィックス）	CU-
後ろにつける文字列（サフィックス）	〈入力されていません〉
採番サイクル	なし
タイムゾーン	日本標準時間
採番キー	〈入力されていません〉

実行結果

『取引先マスタ』アプリ

取引先コード	
64文字以下	
取引先名	担当者名
各務原製作所	斉藤 美波



『取引先マスタ』アプリ

取引先コード	保存ボタンを押すと、自動で取引先コードが採番される。
CU-006	
取引先名	担当者名
各務原製作所	斉藤 美波

まとめ

Excel では、ROW 関数の戻り値と任意の文字列を組み合わせで自動採番を行うケースが見受けられます。ROW 関数で返される行番号に近いものとして、kintone にはレコード番号がありますが、レコード番号は kintone の計算フィールドの計算式で用いることができません。そのため、今回の例にある「CU-001」のような文字列の生成に使用することはできませんが、[やること : [自動採番を行う](#)] を用いれば、柔軟に採番ルールを決定することができます。(課題：レコード番号は計算フィールドの計算式に使用できない)

なお、[やること : [自動採番を行う](#)] を使用すれば同じ番号が取得されることはありませんが、登録されるデータ自体が kintone 上で重複していないかをチェックすることはできません。そのため、そういったチェックをした場合は、kintone の基本機能の「重複を禁止する」を用いて、同じ内容のレコードを二重登録しないように設定しておく必要があります。

ちなみに、既存レコードに対しては Excel で番号をすでに取得しており、新たに登録されたレコードから kintone とカスタマイズで採番したいといった理由で、任意の番号から自動採番を開始したいケースがあるかと思います。その場合は、[やること : [自動採番を行う](#)] だけでは対応できないので、採番キーを用いたり、[やること : [数値をフォーマットする](#)] と組み合わせ、採番結果に数字を足すことで、任意の番号から採番できます。

参考：自動採番を任意の番号から開始することはできますか？

<https://support.gusuku.io/ja-JP/support/solutions/articles/36000218479>

3-2. 入力ミス防止：入力規則（編集不可）

Excel での取引先コードの管理には、もうひとつ頭の痛い問題がありましたので、志摩氏はそれも解決することにしました。

これまでの『取引先マスタ』シートの運用（再掲）

取引先コード	取引先名
CU-001	金都運総研
CU-002	林田商会
CU-003	板橋電子株式会社
CU-004	新山物産
CU-005	岩下税理士事務所

運用：取引先ごとに、取引先コードを手入力またはオートフィルで採番していた。

トラブル内容：取得した取引先コードを誤って上書きしてしまう。

志摩氏のチャレンジ：

「取引先コード」フィールドを編集できないようにしたい

カスタマイズで自動採番ができるようになったので、「取引先コード」フィールドを手入力する必要がなくなりました。

そこで志摩氏は、Excel 時代のトラブルを再発させないために、「取引先コード」フィールドを編集できないようにしたいと考えました。

まず、志摩氏は kintone のフィールドのアクセス権を設定することで、「取引先コード」フィールドを編集できなくする方法を試してみました。すると、思惑通り「取引先コード」フィールドが無効化されたので、これで課題が解決したかと思われました。

『取引先マスタ』アプリ

フィールドのアクセス権 [?ヘルプ](#) | [便利に使おうガイドブック vol.08 アクセス権編](#)

フィールドごとに、そのフィールドを閲覧・編集できるユーザーを制限できます。

フィールド	ユーザー/組織/グループとアクセス権				
取引先コード	<p>ユーザーを追加 <input type="text"/> または <input type="button" value="フォームのフィールドを追加"/></p> <p>ユーザー/組織/グループ</p> <table border="1"><thead><tr><th>ユーザー/組織/グループ</th><th>許可する操作</th></tr></thead><tbody><tr><td>Everyone</td><td><input checked="" type="checkbox"/> 閲覧 <input type="checkbox"/> 編集</td></tr></tbody></table>	ユーザー/組織/グループ	許可する操作	Everyone	<input checked="" type="checkbox"/> 閲覧 <input type="checkbox"/> 編集
ユーザー/組織/グループ	許可する操作				
Everyone	<input checked="" type="checkbox"/> 閲覧 <input type="checkbox"/> 編集				



『取引先マスタ』アプリ

取引先コード	
CU-001	
64文字以下	
取引先名	担当者名
金都運総研	下山 達士
64文字以下	

課題：自動採番の結果を「取引先コード」フィールドに セットできなくなった

ところが、フィールドのアクセス権を設定したため、せっかくカスタマイズで自動採番した結果さえも、「取引先コード」にセットされなくなってしまいました。これでは本末転倒です。志摩氏はフィールドのアクセス権の設定を元の設定前の状態に戻し、今度はカスタマイズで無効化を試みました。

『取引先マスタ』アプリ

取引先コード	
取引先名	担当者名
各務原製作所	斉藤 美波

カスタマイズしてみた

2	
フィールドを無効化する	追加画面・編集画面を表示した時
フィールド 取引先コード	一覧画面の鉛筆を押しした時 一覧画面の鉛筆を押しした時は含めない

実行結果

『取引先マスタ』アプリ

取引先コード

CU-006

64文字以下

取引先名

各務原製作所

64文字以下

担当者名

斉藤 美波

自動採番の結果はセットされるが、
編集画面では無効化されており、
手入力で編集できないようになった。

まとめ

Excelでの運用の場合、セルへの入力内容をコントロールさせたい時に、入力規則機能を使って編集を制限したり、いっそのこと所定のセルを編集できないように制御する場合があります。ところが実際に運用してみると、制限したはずがされていない、制限しすぎてかえって非効率になってしまうといった課題も生まれやすいのではないのでしょうか。

そのような課題に対して、カスタミンでは [やること : [フィールドを無効化する](#)] を用いれば、任意のフィールドを無効化し、編集できなくすることができます。

kintone のフィールドのアクセス権を使うことで、フィールドの編集ができないように制限することはできますが、以下のような違いがあります。

フィールドのアクセス権を使用する場合：

取引先コードを編集してしまうミスを防ぐことはできますが、いかなる場合でも閲覧しかできないため、カスタマイズで自動採番した結果を取引先コードフィールドにセットすることができません。これは、kintone に登録された画面のカスタマイズの実行は、ログインユーザー（kintone の画面を開いているユーザー）の権限で実行されるため、必ず kintone のアクセス権の設定に従って動きます。またカスタマイズで、このアクセス権の設定を無視したり、飛び越えてレコード操作するようなことはできません。

[やること : [フィールドを無効化する](#)] を使用する場合（フィールドのアクセス権は使用しません）：
編集画面からの編集をできなくするため、ユーザーによる取引先コードの編集は防ぎつつ、自動採番の結果を取引先フィールドにセットすることができます。（課題：自動採番の結果を「取引先コード」フィールドにセットできなくなった）

参考：フィールドにアクセス権を設定する

https://jp.cybozu.help/k/ja/user/app_settings/rights/field_rights.html

参考：Customine に設定する kintone 接続ユーザーの権限・カスタマイズの実行権限について

<https://support.gusuku.io/ja-JP/support/solutions/articles/36000380502>

4. ミッションIII『注文管理』アプリの集計 結果を『取引先マスタ』アプリに反映せよ!

4-1. 集計 : COUNTIFS

現在は関連レコード一覧を用いて、『注文管理』アプリから返却遅延のレコードを表示させることで取引先ごとの遅延件数を把握しています。ですが、関連レコード一覧はそれぞれのレコードを開いて確認する必要があるため、全体の返却遅延回数を俯瞰して見ることはできません。そのため、これまで通り一覧形式で遅延件数を確認したり、返却遅延回数で取引先を絞り込みできるようにしたいと考えました。

これまでの『取引先マスタ』シートの運用

『返却遅延』の数を取引先コードごとに集計

=COUNTIFS(注文管理!\$C\$6:\$C\$12
[取引先コード],A2,注文管理!
\$L\$6:\$L\$12[返却遅延],"返却遅延")

	A	B	C	D	L
1	取引先コード	取引先名	担当者名	所在地	返却遅延回数
2	CU-001	金都連総研	下山 達士	岐阜県岐阜市xxxx	0
3	CU-002	林田商会	森 惇	埼玉県浦和市xxxx	0
4	CU-003	板橋電子株式会社	未永 妃里	神奈川県小田原市xxxx	0
5	CU-004	新山物産	金子 真帆	大阪府大阪市北区梅田xxxx	1
6	CU-005	岩下税理士事務所	佐々木 樹里	兵庫県神戸市須磨区xxxx	1

注文管理シート

C	L
取引先コード	返却遅延
CU-001	
CU-004	返却遅延
CU-003	...
CU-005	返却遅延
CU-003	
CU-002	
CU-001	

運用:『注文管理』の『返却遅延』の数を Excel の COUNTIFS 関数 (以下、COUNTIFS と記載) で、取引先コードごとに集計し、表示している。

志摩氏のチャレンジ：関連レコードをカウントすることで、 遅延返却回数を集計する

志摩氏はすでに作成している関連レコードのうち、返却遅延のレコードの行数をカウントすることで、遅延返却回数を集計できるのではと考え、カスタマイズを行ってみました。

『取引先マスタ』アプリ

取引先コード
CU-001

取引先名
金都運総研

担当者名
下山 達士

所在地
岐阜県多治見市××××

注文履歴

受注日	返却遅延
2023-05-31	
2022-02-24	

返却遅延回数

カスタマイズしてみた

カスタマイズするアプリ：『取引先マスタ』

44

関連レコード一覧の条件でレコードを取得する

関連レコード一覧フィールド 注文履歴

最大取得件数 (入力されていません)

自レコードも条件に一致した場合 自レコードも含めて取得する

詳細画面を表示した時

45

取得したレコードを絞り込む

元のレコード 44

条件になるフィールド 返却遅延

条件 等しい

比較値 返却遅延

他のアクションの実行が完了した時

アクション 44

46

レコード行数をカウントする

レコード選択アクション 45

結果をセットするフィールド (省略可) 返却遅延回数

他のアクションの実行が完了した時

アクション 45

実行結果

『取引先マスタ』アプリ

取引先コード
CU-001

取引先名 担当者名
金都運総研 下山 達士

所在地
岐阜県多治見市××××

注文履歴

受注日	返却遅延
2023-05-31	
2022-02-24	

返却遅延回数
0

「注文履歴」フィールドの「返却遅延」のレコードの行数をカウントしてセットされた

無事、返却遅延回数がカウントされ、指定のフィールドにセットされました。これで事なきを得たように思われましたが、すぐに問題点が見つかりました。

課題：各取引先のレコードを開かないかぎり、返却遅延回数が更新されない

返却遅延回数のカウントは、『取引先マスタ』の詳細画面が開いたときに実行されるように設定しています。そのため、各取引先のレコードを開かないかぎり、返却遅延回数が更新されないままになってしまうのです。

これでは不便であるばかりか、返却遅延回数を誤認させてしまうことになりかねません。

そこで、『取引先マスタ』アプリではなく、『注文管理』アプリの更新を受けて、返却遅延回数をカウントするようなカスタマイズに切り替えることにしました。

カスタマイズしてみた

カスタマイズするアプリ：『注文管理』

40 条件を組み立ててレコードを取得する

取得元アプリ [Excel慣れ]注文管理

検索条件 (返却遅延 in ("返却遅延"))
and (取引先コード = "{\$取引先コード}")

最大取得件数 <入力されていません>

レコードを保存した直後 (削除後は除く)

41 レコード行数をカウントする

レコード選択アクション 40

結果をセットするフィールド (省略可) <選択されていません>

他のアクションの実行が完了した時

アクション 40

42 レコードを更新する (キーの値を直接指定)

更新先アプリ [Excel慣れ]取引先マスタ

キーとなるフィールド 取引先コード

キーの値 =取引先コード

マッピング 返却遅延回数=\$41

更新の機会をチェックする チェックする

他のアクションの実行が完了した時

アクション 41

実行結果

『取引先マスタ』アプリ

取引先コード↑	取引先名	担当者名	所在地	返却遅延回数
CU-001	金都運総研	下山 達士	岐阜県多治見市××××	0
CU-002	林田商会	森 惇	埼玉県浦和市××××	0
CU-003	板橋電子株式会社	末永 妃里	神奈川県小田原市××××	0
CU-004	新山物産	金子		1
CU-005	岩下税理士事務所	佐々木		1

各取引先のレコードを1件ずつ開かなくても、『注文管理』アプリが更新されたタイミングで、『取引先マスタ』アプリの返却遅延回数が更新されるようになった。

まとめ

『注文管理』アプリでレコードが保存されたら、[やること: [条件を組み立ててレコードを取得する](#)] で、集計対象とするレコードを絞り込む条件範囲 (=フィールド) と検索条件 (=値) を指定し、条件に合致したレコードのみを取得します。

[やること : [条件を組み立ててレコードを取得する](#)] の条件ビルダー (検索条件)

条件ビルダー

条件ビルダー

全ての条件を満たす

条件2 返却遅延 次*の*いずれかを含む 返却遅延 挿入 + -

条件1 取引先コード =(等しい) \${取引先コード} 挿入 + -

条件追加 グループ追加 並び順追加

OK キャンセル

そして後続の [やること : [レコード行数をカウントする](#)] でその取得したレコードの数をカウントし、その結果を [やること : [レコードを更新する \(キーの値を直接指定\)](#)] で、各取引先のレコードの返却遅延回数フィールドにセットしています。これで、『注文管理』アプリでのレコード更新時に、『取引先マスタ』アプリの返却遅延回数をタイムリーに更新することができ、『取引先マスタ』アプリの一覧画面上で、常に最新の情報を確認できます。(課題：各取引先のレコードを開かないかぎり、返却遅延回数が更新されない)

ちなみに、[やること : [条件を組み立ててレコードを取得する](#)] で [条件 : [レコードを保存した直後 \(削除後は除く\)](#)] を使用していますが、カスタマイズには他にも [条件 : [レコードを保存する直前 \(削除時は除く\)](#)] という条件があります。[条件 : [レコードを保存する直前 \(削除時は除く\)](#)] だと、エラーチェックなどを行っている場合、今回の例でいうと『注文管理』アプリの保存はキャンセルされたにも関わらず、『取引先マスタ』アプリだけ更新されてしまいます。そういったトラブルを防ぐため、アプリ間の連携を行う場合は、必ず [条件 : [レコードを保存した直後 \(削除後は除く\)](#)] を使用します。

おさらい : COUNTIFS 関数

関数の構文

COUNTIFS (条件範囲 1, 検索条件 1, [条件範囲 2, 検索条件 2],...)

関数の説明

複数の範囲のセルに条件を適用して、すべての条件が満たされた回数をカウントします。

引用

<https://support.microsoft.com/ja-jp/office/countifs-%E9%96%A2%E6%95%B0-dda3dc6e-f74e-4aee-88bc-aa8c2a866842>

コラム：試行錯誤の繰り返し

志摩氏は最終的にこのカスタマイズに行きつくまでに、以下のようなカスタマイズを検討していました。

1. まずは、サポートサイトで見つけた「関連レコード一覧を集計する」を参考にカスタマイズを試してみました。ところがこのカスタマイズでは、詳細画面を開くまで返却遅延回数が最新にならず、要件にあわないと判断し、却下しました。（課題：各取引先のレコードを開かないかぎり、返却遅延回数が更新されない）

2. 次に実は、『取引先マスタ』アプリの一覧画面を表示したときに、全レコードの返却遅延回数を計算しセットするカスタマイズを試していました。ところがこちらでは、一覧画面を開くたびにカスタマイズが動作するため、API リクエスト数が増えてしまったり、全レコードに反映がされる前に画面遷移を行ってしまい処理が途中で終了してしまうなど様々な問題が発生し、こちらの案も却下しました。

同じような結果にたどり着くにも、何通りものカスタマイズ方法が考えられます。色々なカスタマイズを試しながら、業務と要件に合うカスタマイズ方法を見つけていきましょう。

kintone でできることの幅がひろがった志摩氏は、他業務についてもさらなる業務改善を推進していくのでした。

ご紹介したルックアップやレコードの取得・装飾について詳しく知りたい方は、他の『カスタマイズの歩き方』もあわせてぜひご覧ください。

コラム：Excel 業務をアプリ化するにあたって

kintone は Excel の代用品ではないので、現在 Excel で行っている業務を kintone に載せ替える場合は

kintone の特性を理解し、それを活かすデータ構造や運用を検討することが大切です。

また、kintone の基本機能の特性や制限事項をよく理解することが重要です。やみくもにカスタマイズを導入するのではなく、基本機能で十分に役目を果たせる場合は、基本機能で実装することをまずは検討しましょう。

5. 今回使用したやることまとめ

	Excel関数・機能	やること
2-1.	VLOOKUP	他アプリのルックアップを取得しなおす
		フィールドを有効化する
		条件を組み立ててレコードを取得する レコードを書き出す
2-2.	EDATE	日付を計算する
2-3.	条件付き書式	フィールド背景色を変更する フィールド文字色を変更する フィールドの文字を太字にする
2-4.	SUMIF	一覧の条件でレコードを全件取得する レコード内の条件に合う行のフィールド合計値を計算する 数値をフォーマットする メニュー位置に文字を表示する
3-1.	オートフィル (ROW)	自動採番を行う フィールドを無効化する
3-2.	入力規則	フィールドを無効化する
4-1.	COUNTIFS	条件を組み立ててレコードを取得する レコード行数をカウントする レコードを更新する (キーの値を直接指定)

※この対応表は本冊子のシナリオにおける例を示したもので、必ずしも該当の Excel 関数を同じように置き換えられるわけではありません

6. 知っておいていただきたい、 カスタマイズの基礎知識

6-1. 本書で使用する用語

本書ではカスタマイズの画面や用語を以下のように記載します。



アクション

カスタマイズの設定における 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

カスタマイズ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

カスタマイズ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。

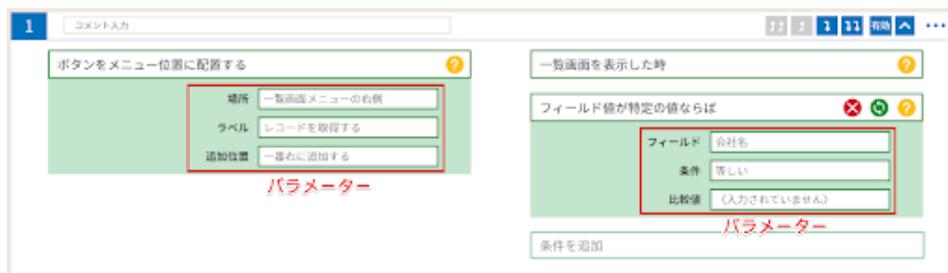
アクション番号

カスタマイズ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。



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。



6-2. カスタマイズの基本的な構造

カスタマイズの基本的な構造は「～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 (<https://docs-customine.gusuku.io/ja/conditions/>) 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。

6-3. カスタミンで作成したカスタマイズが動く仕組み

カスタミンで作成した「kintone アプリのカスタマイズ」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。

※本書に登場するカスタマイズでは使用しませんが、カスタミンの機能に含まれる Job Runner (定期実行 /kintone アプリの Webhook) という機構はユーザーの画面操作とは独立してサーバー側で動く処理となります。以下の説明は Job Runner の処理は考慮していないものとしてお読みください。

6-3-1. 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。

一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：一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せんが、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ませんが、カスタミ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。

6-3-2. 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。

☐ 文字列（1行）の設定

? ヘルプ

フィールド名 *

文字列 (1行)

フィールド名を表示しない

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_

キャンセル 保存

6-3-3. 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

カスタマイズ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、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）

<https://support.gusuku.io/ja-JP/support/solutions/articles/36000268083>

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。カスタマイズ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。

6-3-4. 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>

カスタマイズにおける「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：レコード一覧画面表示後

↓

[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

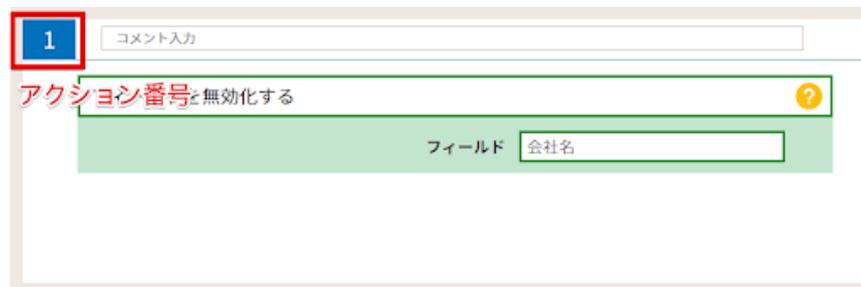
などがあり、例えば「フォームブリッジでデータが登録されたことを契機にカスタマイズを動かしたい」という要件は、「2. イベントが発生しない場合」に該当するためカスタマイズ動きません。

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。

6-4. 実行順序

カスタマイズ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。



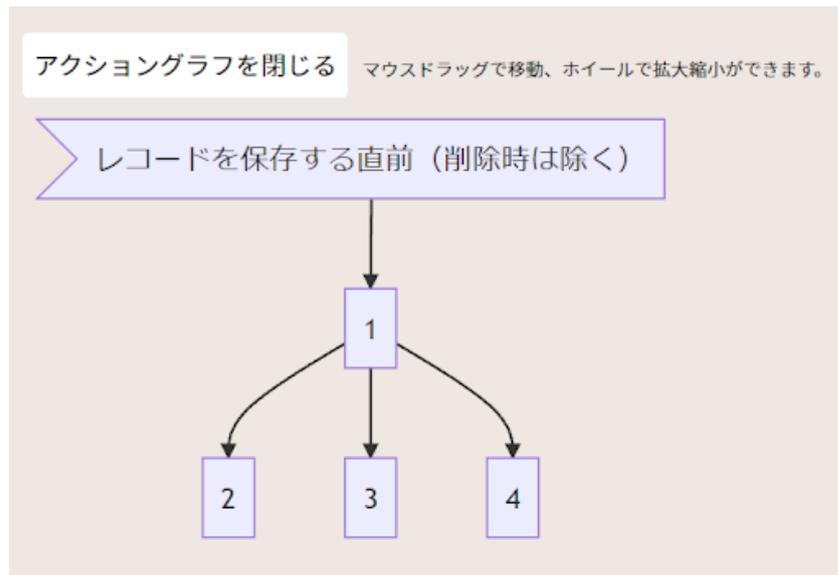
全てのアクションは何かの「イベント」を起点に処理が開始され、「他のアクションの実行が完了した時」や「確認・入力ダイアログで「OK」を押した時」などで繋がっている順に処理されます。

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。



案件管理のカスタマイズ(注) ページ

1 コメント入力

確認ダイアログを表示する

メッセージ入力

OKボタンの名前 OK

キャンセルボタンの名前 キャンセル

レコードを保存する直前（削除時は除く）

2 コメント入力

必須チェックを行う

フィールド A

エラーメッセージ 必須です。

他のアクションの実行が完了した時

アクション 1

3 コメント入力

必須チェックを行う

フィールド B

エラーメッセージ 必須です。

他のアクションの実行が完了した時

アクション 1

4 コメント入力

必須チェックを行う

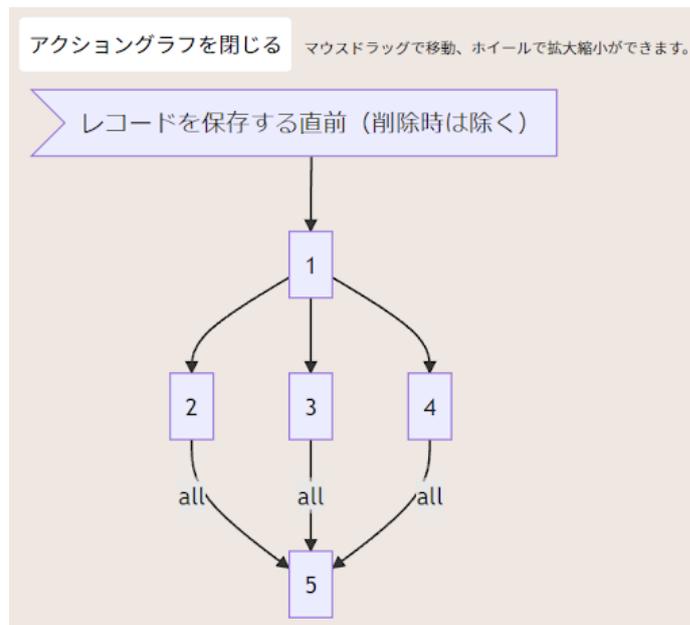
フィールド C

エラーメッセージ 必須です。

他のアクションの実行が完了した時

アクション 1

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。



案件管理のカスタマイズ (1/1) ページ

1	コメント入力	確認ダイアログを表示する	レコードを保存する直前（削除時は除く）
		メッセージ入力 <input type="text"/> OKボタンの名前 <input type="text" value="OK"/> キャンセルボタンの名前 <input type="text" value="キャンセル"/>	
2	コメント入力	必須チェックを行う	他のアクションの実行が完了した時
		フィールド <input type="text" value="A"/> エラーメッセージ <input type="text" value="必須です。"/>	アクション <input type="text" value="1"/>
3	コメント入力	必須チェックを行う	他のアクションの実行が完了した時
		フィールド <input type="text" value="B"/> エラーメッセージ <input type="text" value="必須です。"/>	アクション <input type="text" value="1"/>
4	コメント入力	必須チェックを行う	他のアクションの実行が完了した時
		フィールド <input type="text" value="C"/> エラーメッセージ <input type="text" value="必須です。"/>	アクション <input type="text" value="1"/>
5	コメント入力	フィールドに値を設定する	他のアクションの実行が完了した時
		フィールド <input type="text" value="姓"/> 値 <input type="text" value="完了"/>	アクション <input type="text" value="2,3,4"/>

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。

案件管理のカスタマイズ (2/2) ページ②

6

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

フィールド

7

詳細画面を表示した時

8

フィールド

いずれかのアクションの実行が完了した時

アクション

フィールド値が特定の値ならば

フィールド

条件

比較値

9

フィールド

いずれかのアクションの実行が完了した時

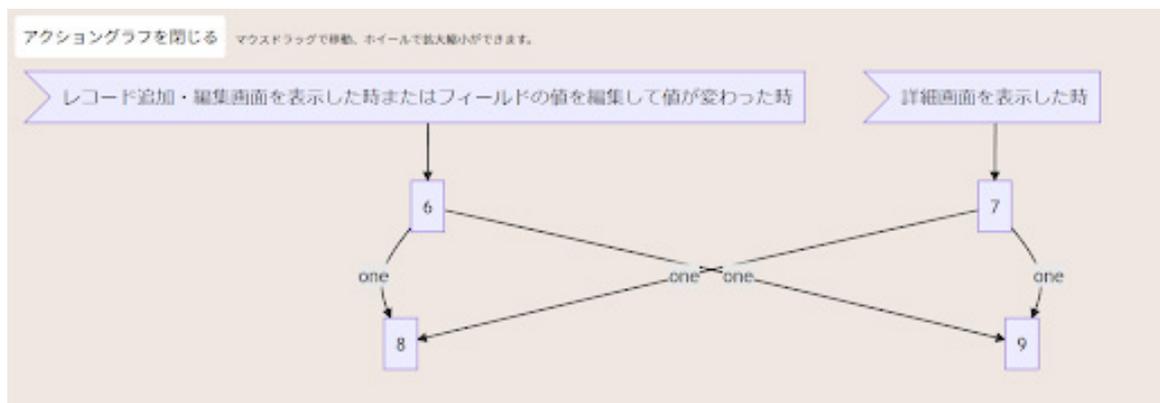
アクション

フィールド値が特定の値でないならば

フィールド

条件

比較値



7. 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、便利なカスタマイズの機能について紹介します。

7-1. 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

カスタマイズのカスタマイズは複数のアクションを組み合わせることで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。

7-2. 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となります。

カスタマイズのほかに 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ズ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「[外部サービス連携](#)」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、カスタマイズ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。

7-3. 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。

カスタマイズのカスタマイズ画面で表示されるエラー

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も、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。



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、カスタマイズ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、カスタマイズ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： [ロックアップを取得しなおす](#)] 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。



このようなエラーが表示されたら、ドキュメントの「制限事項」の項目を確認してみてください。
[やること： [フィールド値をまとめてセットする](#)] の制限事項は下記のように記載されています。

制限事項

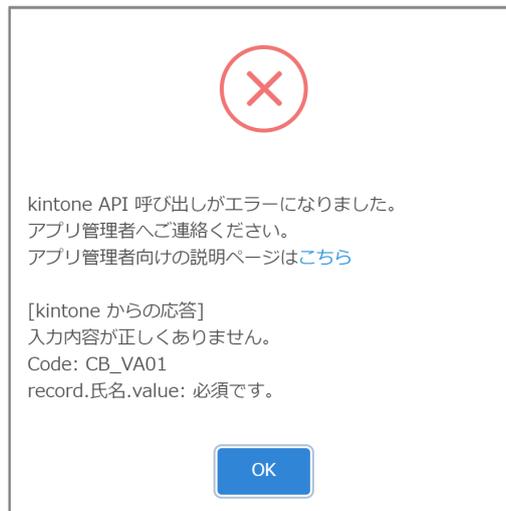
レコード追加画面、編集画面のみで使用可能です。そのほかの画面で使うとエラーが表示されます。

テーブル内のフィールドには対応していません。

[フィールドマッピングの制限事項](#)に記載されているフィールドには対応していません。

今回のエラーは、[やること： [ロックアップを取得しなおす](#)] 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ズ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、[サポートサイト](#)では、カスタマイズ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

(カスタマイズの設定画面右上「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。



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\[0\].〇〇.value: 必須です](#)」というエラーが出ました。詳しく記載していますので、ぜひ確認してみてください。

- [「kintone API呼び出しがエラーになりました」という場合の対応方法](#)
<https://support.gusoku.io/ja-JP/support/solutions/articles/36000243836>
Customineを使ったカスタマイズで、「kintone API呼び出しがエラーになりました」というエラーダイアログが出ることがあります。これを解決するには、kintone側・Customine側...
- [フィールドのエラーをクリアする](#)
https://docs-customine.gusoku.io/ja/actions/error_check/clear_field_error/
概要 フィールドのエラーをクリアします。もともとエラーがない場合は何も起こりません。使い方 「フィールドエラーを表示する」などフィールドにエラーを設定する「やること」を実行したときや、kinton...
- [「入力内容が正しくありません。records\[0\].〇〇.value: 必須です」というエラーが出ました。](#)
<https://support.gusoku.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...

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。

7-4. 原因究明に役立つカスタマイズの便利な機能

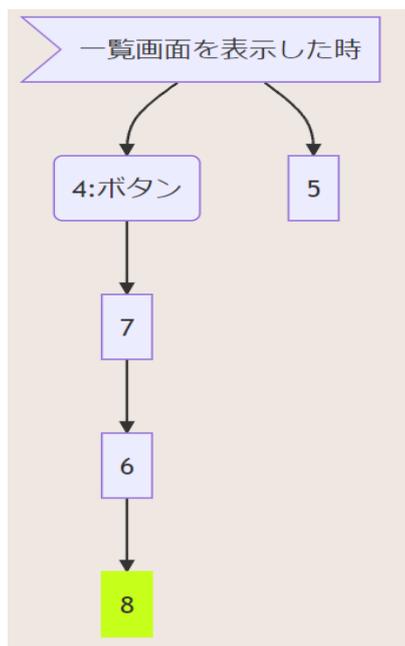
カスタマイズは、うまく動かないときの原因究明に役立つ機能を提供しています。

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。

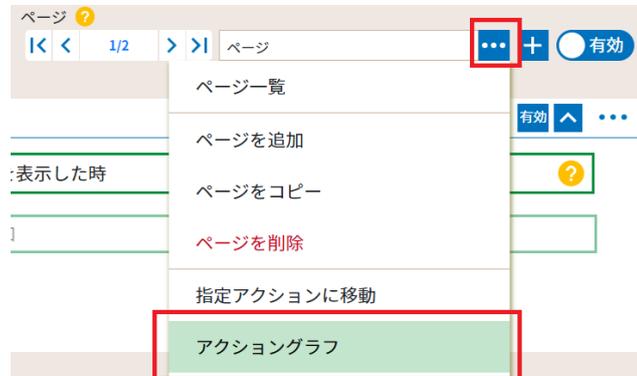
図でアクションの関連性を見て確認できるので、アクションが想定通りにつながっているかを確認したい場合に便利です。

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」を選択すると表示されます。



アクションの無効化

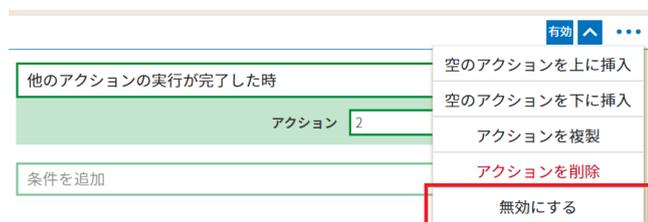
一時的にアクションが実行されないようにしたいときに使用します。

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。



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。



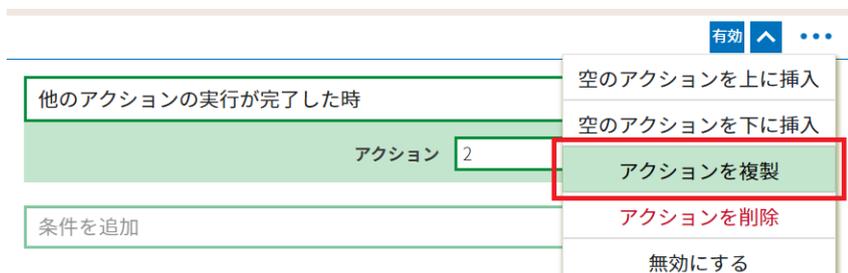
無効化したアクションは、左上のアクション番号がグレーになり、右上には無効と表示されます。



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。



ページの無効化

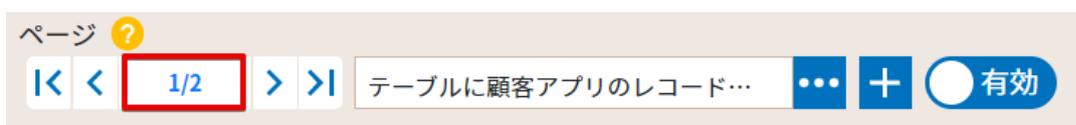
個別のアクションではなく、ページをまるごと無効化・有効化することができます。

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。

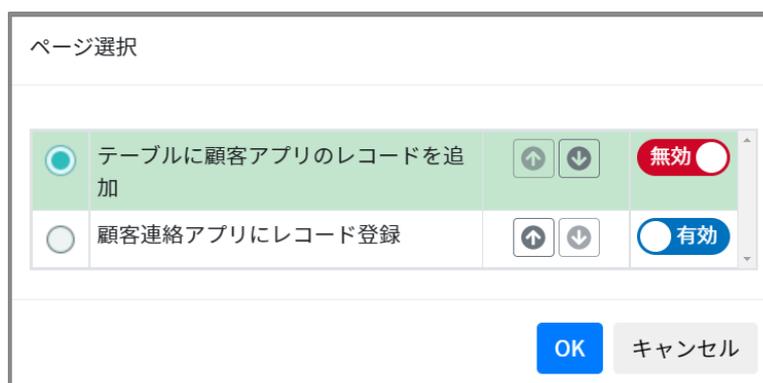
ページの無効化は、カスタマイズ画面右上の 有効 をクリックすると無効に切り替えできます。



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。



ページメニューから無効化することもできます。



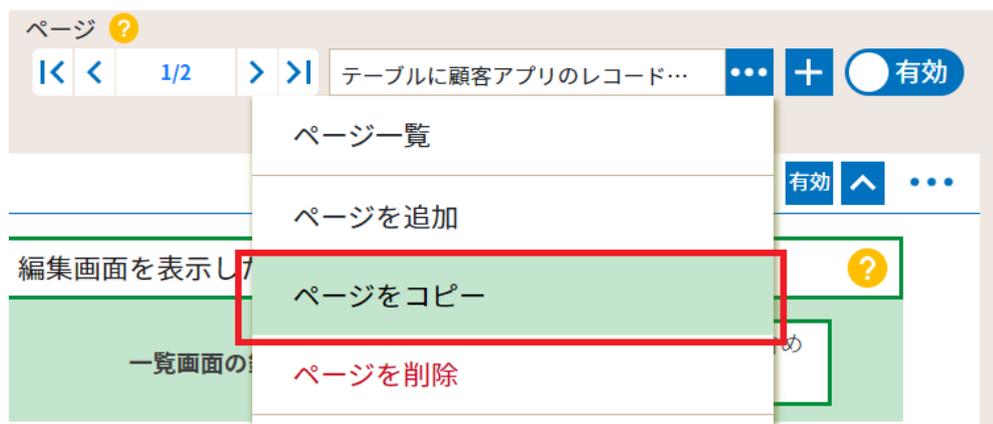
ページの無効化は、他のページの動作を確認したいときに、一旦ページ全体を無効化しておきたい場合などに使用します。この機能を効果的に使うためには、機能ごと、画面ごとにこまめにページを分けてカスタマイズを作成しておくことが必要です。

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。



7-5. 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 : [キーを指定してレコードを取得する](#)] や [やること : [クエリで条件を指定してレコードを取得する](#)] のように、条件を指定してレコードを取得するやんことを使用した場合や、取得したレコードを [やること : [取得したレコードを絞り込む](#)] 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、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。

The image shows two screenshots of a software interface. Screenshot 8 shows a '取得したレコードを絞り込む' (Filter obtained records) dialog box. It has a '元のレコード' (Original record) field with the value '7'. Below it are fields for '条件になるフィールド' (Field to be used as condition) with the value '市町村名' (Municipality name), '条件' (Condition) with the value '含まれる' (Included), and '比較値' (Comparison value) with the value '本町' (Town). To the right is a '他のアクションの実行が完了した時' (When other actions are completed) dialog box with an 'アクション' (Action) field containing '7' and a '条件を追加' (Add condition) button. Screenshot 9 shows a 'レコードの一覧をポップアップで表示する' (Display record list in popup) dialog box. It has a 'レコード' (Record) field with the value '8'. Below it are fields for '表示するフィールド' (Fields to be displayed) with the value '都道府県名, 市町村名' (Prefecture name, Municipality name), '選択方法' (Selection method) with the value '単一選択 (ラジオボタン)' (Single selection (radio button)), 'メッセージ入力' (Message input) with the value '(入力されていません)' (Not input), 'OKボタンの名前' (OK button name) with the value 'OK', 'キャンセルボタンの名前' (Cancel button name) with the value 'キャンセル', '未選択を許可するかどうか' (Whether to allow unselected), with the value '未選択でもOKを許可する' (Allow OK even if unselected), and 'テーブルの幅 (ピクセル)' (Table width (pixels)) with the value '(入力されていません)' (Not input). To the right is a '他のアクションの実行が完了した時' (When other actions are completed) dialog box with an 'アクション' (Action) field containing '8' and a '条件を追加' (Add condition) button.

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	市町村名
<input type="radio"/>	静岡県	川根本町
<input type="radio"/>	大阪府	島本町
<input type="radio"/>	奈良県	田原本町
<input type="radio"/>	和歌山県	串本町
<input type="radio"/>	島根県	川本町

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。

7-6. 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。



Excel 関数と
カスタマイズ編