

「Regexとは何ですか？」をご覧ください

概要

正規表現（別名「regex」を使用すると「WalkMeコンテンツに表示されるダイナミックテキストや自動処理の一部を操作することができます」Using [正規表現の構文](#)は使用しているダイナミックテキストの値に、必要な値と関連のない他の文字の両方が含まれている場合に使用するのに最適です。

ユースケース

Regexのユースケースには以下のようなものがあります。

- ダイナミックテキストにユーザーの姓名が含まれている場合「Regexを使ってファーストネーム（名）のみを選択してShoutOutやバルーンに表示することで、よりパーソナルな表示が可能になります（視覚的な表示）。
- ユーザーのメールアドレス（追加データを含む大きなオブジェクトの一部である場合）を、ページ読み込み時にテキストフィールドに自動入力（自動化）します。
- ダイナミックテキストを使用してユーザーを別のページに自動的に移動させる場合「URLの一部のみを選択します。

ユースケース例

2019年1月31日以前または同等の画面要素（日付を示す）がSmartTip表示されている場合、お客様はSmartTip表示を希望します。

問題への取り組み方：

1. マッチを希望する可能性のある値とマッチを希望しない値の概説から始めます。
2. 複数のルールに問題を分割し、組み合わせた場合、マッチするすべての値を網羅します。

1. 上記のケースでは、マッチさせたい4つのパターンがあります：

- i. 2019年1月の任意の日付
- ii. 2010年11月1日から2018年12月31日までの任意の日付
- iii. 2000年1月1日から2009年12月31日までの任意の日付
- iv. 0000年1月1日から1999年12月31日までの任意の日付

3. 上記のケースごとに処理するルールを形成し、テストします。

- i. `^Jan[\s]+[0123]?[\d]+,[\s]+2019$`

ii. `^\w+\s+[0123]?\d+,\s+201[012345678]$`

iii. `^\w+\s+[0123]?\d+,\s+200[0123456789]$`

iv. `^\w+\s+[0123]?\d+,\s+[01]\d{3}$`

4. 各グループをテストした後、パイプを使って1つの文字列に結合します（余分な^と\$sを削除します）

i.

```
^Jan\s+[0123]?\d+,\s+2019[\w+\s+[0123]?\d+,\s+201[012345678]][\w+\s+[0123]?\d+,\s+200[0123456789]][\w+\s+[0123]?\d+,\s+[01]\d{3}$
```

5. SmartTipのディスプレイセグメントルールセットで上記のRegExを導入：

OSE/jQuery > テキストは正規表現単位 >

```
^Jan\s+[0123]?\d+,\s+2019[\w+\s+[0123]?\d+,\s+201[012345678]][\w+\s+[0123]?\d+,\s+200[0123456789]][\w+\s+[0123]?\d+,\s+[01]\d{3}$
```

6. 結果：要素のテキスト値が0000年1月1日から2019年1月31日までの場合SmartTipが表示されます！

使用するRegexのフォーマットについて

正規表現自体を含む開始のregexタグで始まり、次に正規表現を適用したい文が続き、終了のregexタグで終わります。

[regex=" {regexそのもの} "]{regexを実行するテキスト}[/regex]

注: {正規表現を実行するテキスト}には、複数のダイナミックテキスト値（変数、クッキーWalkMeデータjQueryなど）と静的な値を混在させることができます。Regexにおけるテキストの使用は、すべてではないにしてもほとんどが動的なもので、静的なテキスト値の使用は最小限です。

構造

1. **デマーカーの開始と終了**：括弧以外の場合、^はRegExの開始を表し、RegExの終了を表すために\$で使用

例：

- 拡張パッケージでは`^https://ecolab\.my\.salesforce\.com.*|^https://ecolab.*na.*\visual\.force\.com.*`
- ルールエンジンでregexを使用する場合：ユーザーデータ > 変数 > 部門 > 正規表現単位 > `^sales$`

2. **単一文字マッチ**RegExのほとんどの基本形式は直接文字マッチを行う

例：

- `^walkme$`
 - WalkMeとマッチ

3. **リストと範囲**: `[]` は値のリストを指定するために使用でき、`-` と組み合わせると、ターゲット文字列内の対応する文字が取ることができる値の範囲を指定できます

例：

- `^[dcr][ao][gt]$`
 - 「犬」「猫」「ネタ」「ラット」とマッチ
 - 最初の文字はd,c,r 2文字目はaまたはo 3文字目はgまたはtでなければなりません
 - `^[a-z][0-9]$`
 - 最初の文字はz (小文字) で、2文字目は0から9でなければなりません

4. **繰り返し** : `{3}` および `{#,#}` は RegEx 内で繰り返し同じパターンを入力する必要がないようにするために使用できます

例：

- `^[a-z]{3}$`
 - `[adf]` や「kzd」などの3文字の小文字にマッチします
 - `^[a-z]{1,3}$`
 - `[aam]` や「abc」などの先行文字のうち3文字以下にマッチしますが、「ダクト」はマッチしません

5. **ワイルドカード** : ターゲット文字列内のテキストが動的である場合やコンテンツが問題でない場合に便利です。 `.` 任意の1文字に一致し、`*` は先行するパターンの0個以上に一致し、`+` は先行するパターンの1つ以上に一致します。

例：

- `^.*$`
 - 0個以上の任意の文字に一致し、一致する(サブ)ドメインのみに興味があるため RegEx (特に拡張機能内) の先頭と末尾に追加すると便利です
 - `^a+b*c+$`
 - `+` は、前の文字の少なくとも1つが存在する必要があることを指定するため、マッチします

6. **エスケープ文字** : バックスラッシュ `/` は、それ以降の文字に特別な意味を持つことを示します

例：

- `\.`
 - ワイルドカードではなく期間を表します
 - `/, $` など
 - `/` と `$` にマッチします

- \dと/D
 - 桁\桁なし文字にマッチ
- \sと/S
 - 空白\空白以外の文字にマッチ
- \wと/W
 - 英数字\英数字以外の文字にマッチ

7. 除外文字： [] ^に含まれる場合、ターゲット文字列に文字が存在しないことを表します

例：

- `^[^abc]$`
 - `a[]b[]c`以外の単一文字にマッチします
- `^[^n-p]$`
 - `n[]o[]p`以外の単一文字にマッチします
- `^\w[^og]{2}$`
 - 蜂と海にマッチしますが、ログ、沼、霧などはマッチしません

8. 負の先読み： ?) は、マッチに次の文字列が存在しないことを指定し、たとえばWWalkMe拡張が間違ったiFrameに注入しないようにするのに便利です。

例：

- `^.*eu1\.visual\.force\.com\/(?!Console).*$`
 - eu1.visualforce.com ドメインにマッチしますが、 eu1.visualforce.com/Console はマッチしません

9. パイプ： |はRegexの論理的な「OR」演算子で、複数のパターンにマッチできるターゲット文字列を指定します。

例：

- `^.*ctmstech\.roche\.com\/eclinical.*|.*ctms-performance\.roche\.com.*|.*siebelctmsqa\.kau\.roche\.com.*$`

Regexが使用できる場所

Regexは以下のステップタイプとWalkMeアイテムのBBCodeに挿入することができます。

- 視覚表示
 - スマートウォークスルーステップ/ポップアップ;
 - ShoutOut;
 - スマートチップ (ガイダンスタイプ)
 - Survey
- 自動化。
 - スマートウォークスルー自動ステップ (テキストを塗りつぶし & リスト値のステップタイプ)

- プを選択);
- ウォークスルーリダイレクトステップ。

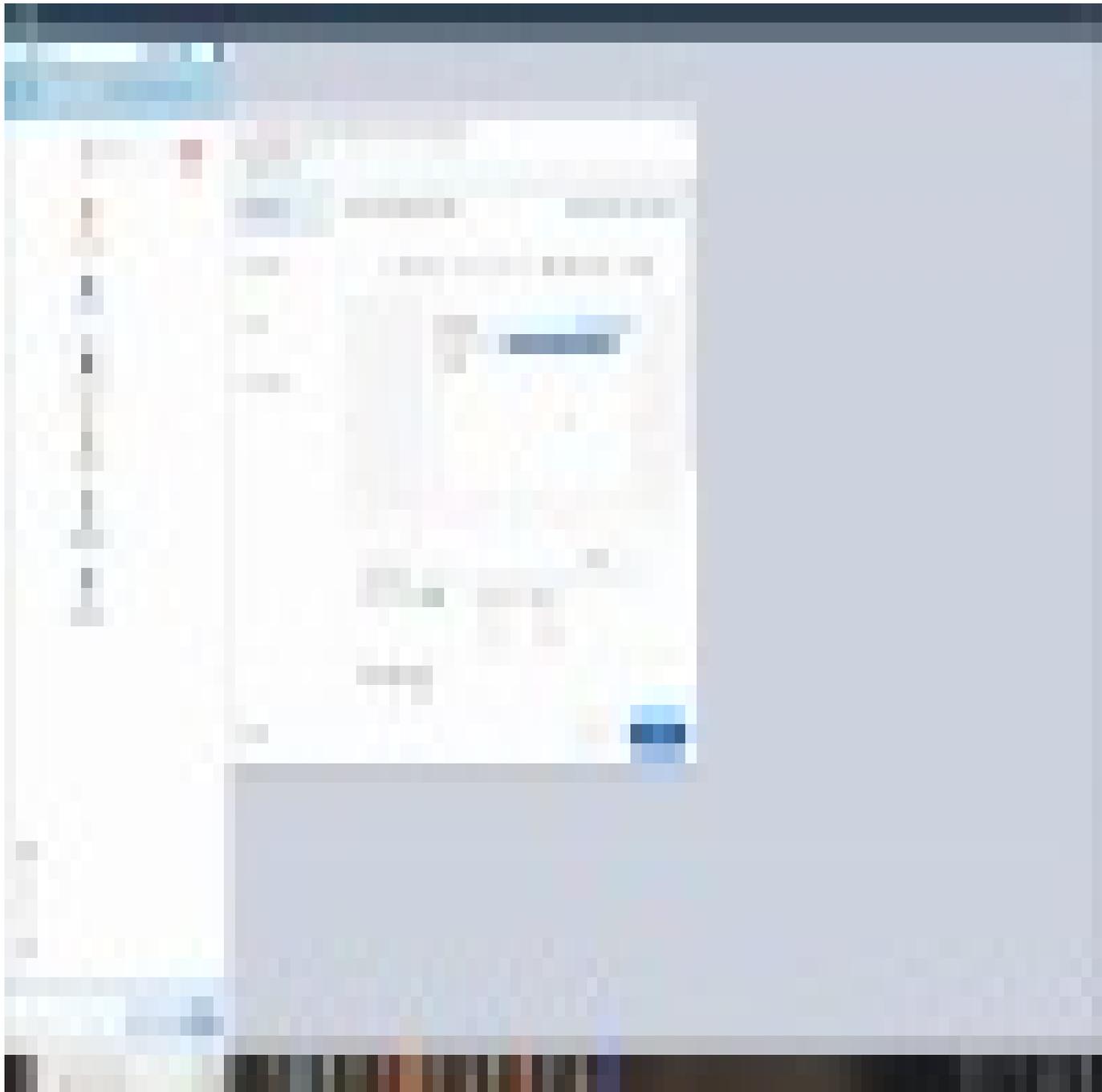
WalkMeのレゲックス

レゲックスは、次のWalkMeインスタンスで使用できます：

1. 次のルールタイプのルールエンジン：
 - 選択したエレメント
 - URLの基になるURLルールは、
 - ユーザー属性
 - jQuery
 - 画面上エレメント
 - 変数（ユーザーデータ）
2. BBCode

Tip Tuesday動画

WalkMeでのChatGPT for Regexの使用



会話に参加

ChatGPTの提案をすべて検証することを忘れないでください！ [コミュニティ内のWalkMeコンテンツにOpenAIをどのように活用しているか教えてください](#)□