OpenRefineの機能(ver3.5.2)

表計算ソフトとここが違う

1. マッチした行だけに作用する
2. 再計算しない
3. 違う行のデータにはアクセスできない（レコードモードは抜け道）
4. 行モードとレコードモードがある

メニューだけでできること（マッチしている行だけに作用することに注意）

メニュー	対応GREL	結果（注意点）
セル編集/よく使う変換/数値に	value.toNumber()	
同/文字列に	value.toString()	
同/日付に	value.toDate()	
同/先頭と末尾の空白を削除	value.trim()/strip()	全角空白はやってくれない
同/連続した空白を縮める	value.replace(/\s+/,' ')	複数の半角空白を空白一つにする
同/頭だけ大文字・大文字・小文字	value.toUppercase()など	欧文のみ有効
同/セルをnullに	null	無効値(null)にする
同/セルを空白に	“”	空白文字列になる
同/HTMLをアンエスケープ	value.unescape(“html)	
同/スマート引用符変換	value.replace(/[‘’,‘‹›,]/,\"\\\'\").replace(/[“”«»„]/,\"\\\"\")",	様々なバリエーションの欧文引用符を”に統一する
セル編集/置換	value.replace("長嶋","長島")	文字列を置換する

ただし、どのような効果を生むかはよくチェックすること！

「数値に」（toNumber()）を適用させた場合

表の構造を変える（マッチしている行だけに作用することに注意）

セル編集/多値のセルを分割	縦方向に「レコード化」する	何行になるかは値次第。区切り文字や文字列長で指定できる
セル編集/多値のセルを結合	同一レコードの値を結合する	
カラム編集/複数のカラムに分割	カラム方向に分割する	
カラム編集/複数のカラムに結合	複数のカラムを結合する	Nullはちゃんと無視してくれる 列は離れていても可
行列転置/縦持ち化/1カラム化	１列に「レコード化」する	
行列転置/縦持ち化/２カラム化	カラム名と値の２列にする	いわゆるTidydataにする
行列転置/等間隔横持ち化	等間隔でカラム化する	１データが繰り返し複数行の場合
行列転置/値に応じた横持ち化	最初の値でカラム化する	キーがあるなら一定でなくても可
行列転置/値に応じた横持ち化	付記カラムを使う場合	２列目も欲しい場合
すべて/カラムを編集/並び替え・削除 各カラム/カラム編集/各種移動	カラムを並び替える	例外的に、マッチしていない（表示されていない）データにも影響する
このカラムに基づいてカラムを追加	新しいカラムを作る	GRELで計算した結果を入れるカラムを新設する
URLでカラムを追加	いわゆるスクレイピング	
ソート	並び替え	「永続的に並び替える」を実行しない限り、並び替えは一時的

表示される行（マッチしている行）を選ぶ

ファセット/文字列ファセット	表示行をフィルタリングする	includeで複数項目を選択できる
ファセット/数値ファセット	数値の範囲でフィルタリング	
ファセット/タイムライン	Date型のデータ専用	
ファセット/散布図	複数の数値でフィルタリング	座標データに有用
ファセット/カスタム	GRELで処理したフィルタリング	単語、文字列長、エラーなどでフィルタリング。**重複ファセット**は、重複項目がないかをチェックする
星ファセット・旗ファセット	マニュアルで（不要あるいは要注意な）行を選択する	大抵の場合、最後の手段

GREL（javascriptに似た専用言語）

関数	機能	注意
value.toString()	文字列に変換	数値に対してはtoString(“%.0f”)で整数表現に、日付型に対してはtoString(“yyyy/MM/dd”)なら表記を指定できる
value.chomp(“様”)	最後にあれば削除	最後尾にその文字列があれば削除する
value.substring(3,5)	文字列の一部	3文字目から5文字目**前まで**を抜き出す。**マイナスの数字は語尾からカウントした位置を表す**
value.replace(“様”,”さん”)	文字の置換	**空白に変換することで実質的に削除できる**

そのまま使える正規表現（regex101.comで練習あるのみ）

1．数字以外を空白に変換　value.replace(/[^0-9.+\-]+/, " ")

/[^0-9.+\-]+/

[^...]...以外、0-9は数字全部、[]の中の+や.はそのままの意味、桁記号,を含めるかは状況次第

2. 複数の数字の塊を捕獲　value.find(/[\d.\-+]+/)

/[\d.\-+]+/

数字か+か-か.が１個以上で

3. メールアドレス　value.find(/[\w\-._]+@[\w\-._]+\.[A-z]+/)

/[\w\-._]+@[\w\-._]+\.[A-z]+/

メールアドレスを示すパターン

4. ウェブサイトのURL　value.find(/https?://[\w!?/+\-_~;.,*&@#$%()'[\]]+/)

/https?://[\w!?/+\-_~;.,*&@#$%()'[\]]+/

httpかhttpsで始まるURLパターン

4. 全角日本語　value.find(/[ぁ-んァ-ケｱ-ﾝﾞﾟ一-龠]+/)

/[ぁ-んァ-ケｱ-ﾝﾞﾟ一-龠]+/

欧文の[a-zA-Z]に相当する

クラスタリング　似た文字列を統合する

照合(reconcile)　外部データと照合する（あまり使い道がないので省略）

各ソフトのインストール

Firefoxのアドオン

　右側の三本線のボタンから「アドオンとテーマ」を選び、「アドオンを探す」で「Table to Excel」や「DownThemAll」で検索する。インストールする時、権限を許可する必要がある。ただし、アドオン（Chromeなら拡張機能）の安全性には何の保証もないので、むやみに増やさないこと。

OpenRefine

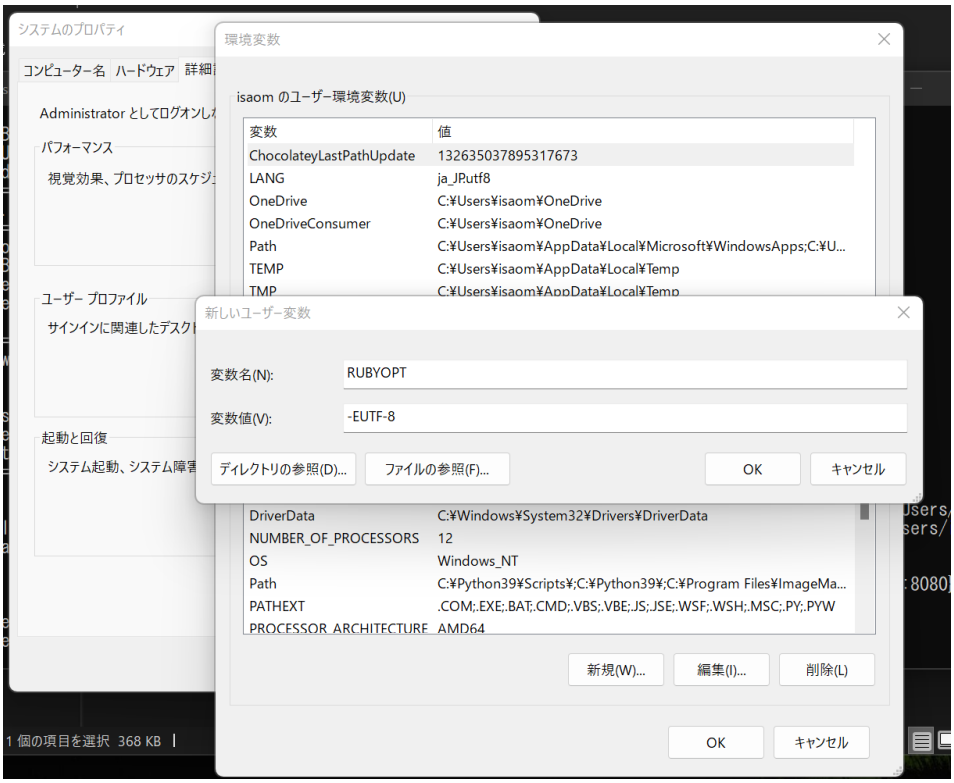
　公式ページ（https://openrefine.org/）のDownloadのページから、OSに合うパッケージをダウンロードする。Windowsの場合は、通常は「Windows Kit」を選ぶ。

Java VM

　OpenRefineやTabulaはJava仮想マシン（VM）という環境で動くので、Javaがインストールされていない場合は動かない。Java VMは（https://www.java.com/ja/download/ie_manual.jsp）からインストールする。会社によっては、ウイルス対策ソフトでインストールできないようになっている場合があるので、担当者に相談する必要がある。

Tabula

　公式ページ（https://tabula.technology/）からOSに合うパッケージをダウンロードする。m
日本語Windowsの場合、ユーザー環境変数RUBYOPTを「-EUTF-8」に設定する必要がある。（TabulaはRubyというプログラム言語で文字コードUTF-8を前提に書かれているので、Rubyにそのことを教える必要がある）



補足情報

文部科学省: 医学部医学科の入学者選抜における公正確保等に係る調査について
(https://www.mext.go.jp/a_menu/koutou/senbatsu/1409128.htm)

ISU: 北京五輪フィギュア結果(https://results.isu.org/results/season2122/owg2022/)

2014年衆院選(https://web.archive.org/web/20141217031821/https://special.jimin.jp/speech/)はテーブル構造

2017年衆院選(https://web.archive.org/web/20171108013923/https://www.jimin.jp/election/results/sen_shu48/speech/)

2021年衆院選(https://web.archive.org/web/20211102025034/https://special.jimin.jp/speech/)はリスト構造

国会議員いちらんリスト(https://web.archive.org/web/20211024071607/https://democracy.minibird.jp/)はテーブル構造。欲しいデータは属性値

OpenRefineでGoogle Geocodingを使う場合のURL構成式

```
"https://maps.googleapis.com/maps/api/geocode/json?address=" +
escape(value, "url")+ "&key=<あなたのグーグルAPIのアクセスキー>"
```

Geocodingの結果から経度/緯度を抜き出す式

```
parseJson(value).result[0].geometry.location.lng + "/" +
parseJson(value).result[0].geometry.location.lat
```