SQL(3)
コーパスデータの扱い
▶演習

CASE式

```
select・・・,
count (case when ○○='' then 1 else 0 end) as ~,
from ・・・
```

○○に列名を入力し，その条件に合う例を
1としてカウントする

CASE式

（演習１）

中世に頻出する「御～有る」という敬語表現の用例数を，ジャンルごとの内訳と併せて知りたい

→午前中の問題１２から

＝＝

長単位表から，「室町時代編」を対象に，
「御～有る」という語を抽出し，
同じ語彙素，語彙素読みのものを一行にまとめて，
さらにキリシタン，狂言それぞれの内訳を示す
（語数の多い順に並べる）

問題12の解答例から

ジャンルごとの内訳を
count関数で示す

```
select L.語彙素,L.語彙素読み,B.ジャンル,count(L.語彙素)
from 長単位 as L inner join 書誌情報 as B
on L.サンプルID=B.サンプルID
where L.語彙素 like N'御%有る'
group by L.語彙素,L.語彙素読み,B.ジャンル
order by B.ジャンル,L.語彙素,L.語彙素読み
```

CASE式

指定した条件のうち，ジャンルが{キリシタン資料／狂言}のものがあれば，それを1としてカウントする

（演習Ⅰ　解答例）

```
select L.語彙素,L.語彙素読み,count(*) as 粗頻度,
count (case when B.ジャンル ='キリシタン資料' then 1 else 0 end) as キリシタン資料 ,
count (case when B.ジャンル ='狂言' then 1 else 0 end) as 狂言資料
from 長単位 as L inner join 書誌情報 as B on L.サンプルID=B.サンプルID
where L.語彙素 like N'御%有る'
group by L.語彙素,L.語彙素読み
order by count(L.語彙素)desc
```

CASE式

```
use chunagon_chj
select L.語彙素,L.語彙素読み,B.ジャンル,count(
from 長単位 as L inner join 書誌情報 as B
on L.サンプルID=B.サンプルID
where L.語彙素 like N'御%有る'
group by L.語彙素,L.語彙素読み,B.ジャンル
order by B.ジャンル,L.語彙素,L.語彙素読み
```

結果 メッセージ

	語彙素	語彙素読み	ジャンル	(列名なし)
1	御上せ有る	オノボセアル	キリシタン資料	1
2	御上り有る	オノボリアル	キリシタン資料	3
3	御下り有る	オクダリアル	キリシタン資料	5
4	御乗せ有る	オノセアル	キリシタン資料	1

```
use chunagon_chj
select L.語彙素,L.語彙素読み,count(*) as 粗頻度,
count (case when B.ジャンル ='キリシタン資料'
then 1 else 0 end) as キリシタン資料 ,
count (case when B.ジャンル ='狂言'
then 1 else 0 end) as 狂言資料
from 長単位 as L inner join 書誌情報 as B
on L.サンプルID=B.サンプルID
where L.語彙素 like N'御%有る'
group by L.語彙素,L.語彙素読み
order by count(L.語彙素)desc
```

結果 メッセージ

	語彙素	語彙素読み	粗頻度	キリシタン資料	狂言資料
1	御免有る	ゴメンアル	28	28	28
2	御語り有る	オカタリアル	26	26	26
3	御出で有る	オイデアル	15	15	15
4	御成り有る	オナリアル	9	9	9
5	御尋ね有る	オタズネアル	8	8	8

ただし，これは左のSQL文で出力した結果を，ピボットテーブルで集計することでも対応できる

CASE式

(演習2)

洒落本に出てくる感動詞について，カタカナ表記のものがどれくらい出現するかを知りたい

＝＝

短単位表から，『江戸時代編』に出現する感動詞について，
・原文文字列にカタカナ表記を含むもの
・カタカナ表記以外のもの
の語数をそれぞれ集計し，
同じ語彙素，語彙素読みのものを一行にまとめて，
語彙素読み順になるよう出力する

※洒落本ではカタカナと一字分の踊り字を対応する仮名に置換
※カタカナ表記の正規表現：［ァ-ヴ］または¥p{Katakana}

CASE式

(演習2)

洒落本に出てくる感動詞について，カタカナ表記のものがどれくらい出現するかを知りたい

＝＝

語彙素	語彙素読み	原文カタカナ表記	原文カタカナ表記以外	合計
	ざっくりしたイメージ…			

CASE式

指定した条件のうち，
原文文字列にカタカナを含む場合
と，含まない場合とで指定している

（演習２ 解答例）

```
select 語彙素,語彙素読み,
sum(case when 原文文字列 like '%[ァ-ヴ]%'
then 1 else 0 end) as 原文カタカナ,
sum(case when 原文文字列 not like '%[ァ-ヴ]%'
then 1 else 0 end) as 原文カタカナ以外,
count(*) as 粗頻度 from 短単位
where サブコーパス名 like '江戸'
and 品詞='感動詞-一般'
group by 語彙素,語彙素読み
order by 語彙素読み
```

CASE式

(演習3)※紹介のみ

CHJに出現する形容詞を
サブコーパスごとに一覧化する

==

短単位表から，品詞が形容詞であるもので，
同じ語彙素，語彙素読みのものを一行にまとめ，
語彙素，語彙素読み，品詞，語種の列と，
各サブコーパスごとの集計を併せて出力する
※試行として，4音節以上（語彙素読みを4文字以上で指定）の形容詞に限定する

CASE式

```
select 語彙素,語彙素読み,品詞,語種,count(*) as 粗頻度,
count (case when サブコーパス名 ='奈良' then 1 else 0 end) as 奈良,
count (case when サブコーパス名 ='平安' then 1 else 0 end) as 平安,
count (case when サブコーパス名 ='鎌倉' then 1 else 0 end) as 鎌倉,
count (case when サブコーパス名 ='室町' then 1 else 0 end) as 室町,
count (case when サブコーパス名 ='江戸' then 1 else 0 end) as 江戸,
count (case when サブコーパス名 ='明治・大正' then 1 else 0 end)
as 明治・大正
from 短単位
where 品詞 like '形容詞%' and len(語彙素読み)>3
group by 語彙素,語彙素読み,品詞,語種
order by 語彙素読み
```

連続する語のデータを取得する

（演習4）

動詞「恐れる」に前接する助詞のバリエーションを，前後文脈付きで知りたい

＝＝

短単位表から，語彙素「恐れる」の直前に出現する品詞「助詞」をキーとして，前後文脈20語ずつのKWICを生成する
サブコーパス名，キーの語彙素，語彙素読み，品詞の列を併せて出力し，品詞・語彙素順に並べる

連続する語のデータを取得する

（演習４　解答例）

```
use chunagon_chj
select s1.サブコーパス名,s1.語彙素,s1.語彙素読み,s1.品詞,
dbo.fn前文脈(s1.サンプルID,s1.出現書字形開始位置,20)
as 前文脈,s1.キー,
dbo.fn後文脈(s1.サンプルID,s1.出現書字形開始位置,20)  as 後文脈
from 短単位 as s1
inner join 短単位 as s2 on s1.サンプルID=s2.サンプルID
and s1.連番 + 10 = s2.連番
where s1.品詞 like '%助詞%' and s2.語彙素 = '恐れる'
order by s1.品詞,s1.語彙素
```

連続する語のデータを取得する

（演習5）

形容詞「良い」に後接する名詞のバリエーションを，前後文脈付きで知りたい

＝＝

短単位表から，語彙素「良い」の連体形の直後に出現する品詞「名詞」の語をキーとして，前後文脈20語ずつのKWICを生成する。
キーの語彙素，語彙素読み，品詞も併せて出力し，
品詞・語彙素順に並べる

連続する語のデータを取得する

（演習５ 解答例）

```
select s1.語彙素,s1.語彙素読み,s1.品詞,
dbo.fn前文脈(s1.サンプルID,s1.出現書字形開始位置,20)
as 前文脈, s1.キー,
dbo.fn後文脈(s1.サンプルID,s1.出現書字形開始位置,20)
as 後文脈
from 短単位 as s1
inner join 短単位 as s2 on s1.サンプルID=s2.サンプルID
and s1.連番= s2.連番 + 10
where s2.語彙素 = '良い' and s2.活用形 like '連体形%'
and s1.品詞 like '名詞%' order by s1.品詞,s1.語彙素
```

連続する語のデータを取得する

（演習6）

平安時代編において，3連続する助動詞のデータを取得したい（N-gram応用として）

＝＝

短単位表から，
『平安時代編』の，3連続する助動詞を対象に，
キーを3つ結合したもの，語彙素ごとに境界（／）を与えたもの，出現頻度を出力し，
出現頻度順に並べる。

連続する語のデータを取得する

（演習６ 解答例）

```
use chunagon_chj
select s1.キー+s2.キー+s3.キー, s1.語彙素+'/'+s2.語彙素+'/'+s3.語彙素,
count(*) as 頻度 from 短単位 as s1
inner join 短単位 as s2 on s1.サンプルID=s2.サンプルID
and s1.連番 + 10 = s2.連番
inner join 短単位 as s3 on s2.サンプルID=s3.サンプルID
and s2.連番 + 10 = s3.連番
where s1.サブコーパス名='平安'and s1.品詞 like '助動詞%' and s2.品詞
like '助動詞%' and s3.品詞 like '助動詞%'
group by s1.キー+s2.キー+s3.キー, s1.語彙素+'/'+s2.語彙素+'/'+s3.語彙素
order by count(*) desc
```

連続する語のデータを取得する

```
select s1.キー+s2.キー+s3.キー,
--キーの結合には+を用いる
s1.語彙素+'/'+s2.語彙素+'/'+s3.語彙素,
--語彙素/語彙素/語彙素の形になるように'/'を含めて結合

count(*) as 頻度 from 短単位 as s1
inner join 短単位 as s2 on s1.サンプルID=s2.サンプルID
and s1.連番 + 10 = s2.連番
inner join 短単位 as s3 on s2.サンプルID=s3.サンプルID
and s2.連番 + 10 = s3.連番
--連続するデータであるようにinner joinを利用する

where s1.サブコーパス名='平安' and s1.品詞 like '助動詞%' and s2.品詞 like '助動詞%'
and s3.品詞 like '助動詞%'
--3語とも品詞が助動詞であるように指定する
group by s1.キー+s2.キー+s3.キー, s1.語彙素+'/'+s2.語彙素+'/'+s3.語彙素
order by count(*) desc
```

その他の関数など

（演習7）

演習2の発展

洒落本の感動詞で，カタカナで表記される語が全体のどれくらいの割合を占めるか知りたい

＝＝

短単位表から，
洒落本に出現する品詞「感動詞」を対象に，
カタカナ表記の感動詞の数／感動詞全体の数，が
列として算出されるように出力する

その他の関数など

（演習7 解答例）

```
select 品詞,
sum(case when 原文文字列 like '%[ァ-ヴ]%'
then 1 else 0 end) as 原文カタカナ,
count(*) as 粗頻度,
CONVERT(float,sum(case when 原文文字列 like '%[ァ-
ヴ]%'then 1 else 0 end))/count(*)
from 短単位
where サブコーパス名 like '江戸'
and 品詞='感動詞-一般'
group by 品詞
```

その他の関数など

```
use chunagon_chj
select 品詞,
sum(case when 原文文字列 like '%[ァ-ヴ]%' then 1 else 0 end) as 原文カタカナ,
count(*) as 粗頻度,
--品詞と原文カタカナの用例数と，感動詞全体の語数を列として出したい
--case when を再び使用

CONVERT(float,sum(case when 原文文字列 like '%[ァ-ヴ]%'then 1 else 0 end))/count(*)
--convert関数により，文字列を数値に変換する
--convert(データ型,文字列) ※float=浮動小数点数
--括弧の中身は，原文カタカナの用例数／感動詞の合計

from 短単位 where サブコーパス名 like '江戸' and 品詞='感動詞-一般'
--おおもとの条件は，江戸時代編に出現する感動詞

group by 品詞
--品詞でまとめる
```

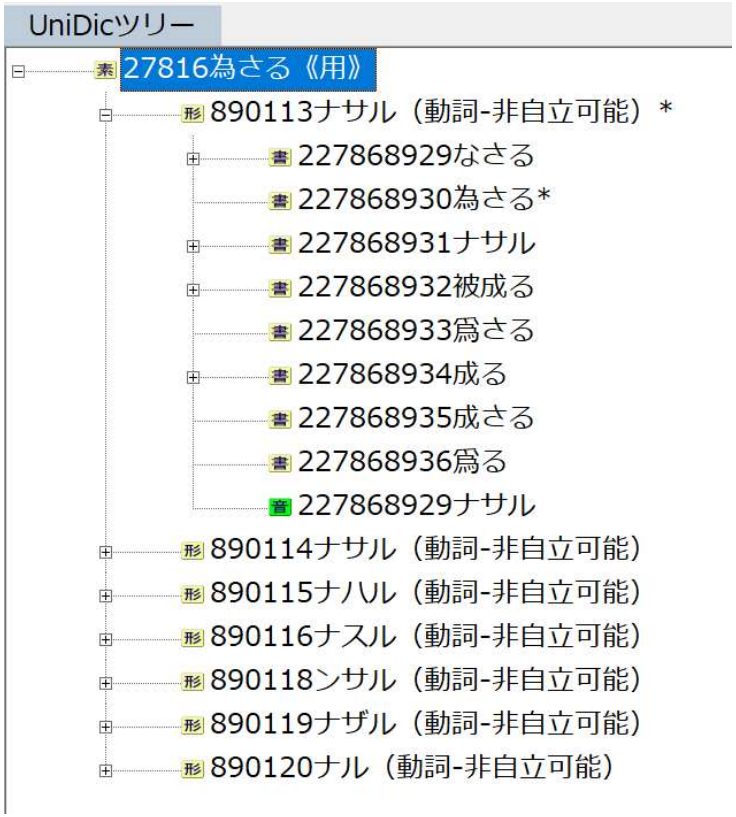
unidicSQLの活用

(演習8)

コーパスと辞書をつなぐ課題の応用として①

==

辞書の見出し表から,
語彙素「為さる(ナサル)」を
対象に,
同じ語彙素,語形,書字形の
ものを一行にまとめ,
語形順に並ぶよう出力する



unidicSQLの活用

（演習８ 解答例）

```
use unidicSQL
SELECT 語彙素,語形,書字形 from 短単位書字形 as O
inner join 短単位語形 as F on F.語形ID=O.語形ID
inner join 短単位語彙素 as L on L.語彙素ID=F.語彙素ID
where 語彙素 like '為さる' and 語彙素読み like 'ナサル'
group by 語彙素,語形,書字形
order by 語形
```

orthtoken ＝ 書字形，form＝語形，lemma＝語彙素

unidicSQLの活用

(演習9)

コーパスと辞書をつなぐ課題の応用として②

==

短単位表から「秋（アキ）」で始まる語彙素で，
かつCHJで使用されているものを集計し，
同じ語彙素，語彙素読み，サブコーパス名のものを
一行にまとめ，サブコーパス順に並ぶよう出力する

unidicSQLの活用

(演習9 解答例)

```
use unidicSQL
select L.語彙素,L.語彙素読み,S.サブコーパス名,count(S.キー)as 粗頻度
from 短単位語彙素 as L
※ inner join chunagon_chj.dbo.短単位 as S on L.語彙素ID=S.語彙素ID
where L.語彙素 like '秋%' and L.語彙素読み like 'アキ%'
group by L.語彙素,L.語彙素読み, S.サブコーパス名
order by S.サブコーパス名
```

※leftにすると，CHJに用例のない語も一覧に含めることができる