

全商情報処理検定試験 1 級ビジネス情報部門 SQL 分野指導法

はじめに

インターネットの検索サイトでキーワードを入力すると、瞬時に関係ホームページリストを応えてくれる。この動作の裏で、実は SQL データベースが働いている。また SQL は、企業の会計や顧客情報などの業務情報管理など実務で幅広く活用されている技術である。

また、データのバックアップ・復元機能やセキュリティ対策を備えているのもデータベースソフトとして大切な機能である。実際に使う気になれば、フリーソフト等で手軽かつ安く導入できる技術ですから、ぜひマスターしよう。

1 SQL 文の基本

SQL 文は、大量のデータを高速に集計するために決まった方式で命令を出す。独特の動作をするので、以下にデータベースの操作例を示しながら、SQL の原理と命令の使い方を説明していく。

(1) SELECT 文の分解

例 SELECT 仕入先, 日付 FROM 仕入伝票表 WHERE 商品コード = '2'

列 (フィールド) を選び出す。つまり射影 (projection) の操作。射影は、必要な列を目的の順序に並べる。列の順序が逆でも、繰り返してもかまわない。また、計算式にしても表示される。SQL では、* ですべての列を出力する。

行 (レコード) を選び出す。つまり選択 (select) の操作。選択の操作で、不必要な行は封じ込まれ、必要な情報だけに絞り込むことができる。

仕入伝票表

| 通し番号 | 日付 | 仕入先 | 商品コード | 数量 | 単価 |
|------|----------|-------|-------|----|------|
| 1 | 20080401 | 国際電器 | 2 | 2 | 3200 |
| 2 | 20080408 | YH 商事 | 1 | 4 | 9400 |
| 3 | 20080412 | テレホープ | 1 | 7 | 9200 |
| 4 | 20080414 | YH 商事 | 2 | 8 | 5650 |
| 5 | 20080427 | 国際電器 | 1 | 6 | 8800 |

SELECT 文の表示結果

| 仕入先 | 日付 |
|-------|----------|
| 国際電器 | 20080401 |
| YH 商事 | 20080414 |

(2) 表の結合動作

例 SELECT * FROM 仕入伝票表, 商品マスタ表

仕入伝票表

| 通し番号 | 日付 | 仕入先 | 商品コード | 数量 |
|------|----------|-------|-------|----|
| 1 | 20080401 | 国際電器 | 2 | 2 |
| 2 | 20080408 | YH 商事 | 1 | 4 |
| 3 | 20080412 | テレホープ | 1 | 7 |

商品マスタ表

| 商品コード | 商品 |
|-------|-------|
| 1 | 液晶テレビ |
| 2 | デジカメ |

2つの表のフィールド名7つがすべて取り上げられる。同じフィールド名は表、フィールド名と明示して区別する。

SELECT 文の表示結果

| 通し番号 | 日付 | 仕入先 | 仕入伝票表. 商品コード | 数量 | 商品マスタ表. 商品コード | 商品 |
|------|----------|-------|--------------|----|---------------|-------|
| 1 | 20080401 | 国際電器 | 2 | 2 | 1 | 液晶テレビ |
| 1 | 20080401 | 国際電器 | 2 | 2 | 2 | デジカメ |
| 2 | 20080408 | YH 商事 | 1 | 4 | 1 | 液晶テレビ |
| 2 | 20080408 | YH 商事 | 1 | 4 | 2 | デジカメ |
| 3 | 20080412 | テレホープ | 1 | 7 | 1 | 液晶テレビ |
| 3 | 20080412 | テレホープ | 1 | 7 | 2 | デジカメ |

SQL はとりあえずすべての組合せを自動的に作り出してしまふ。レコード数は2つの表のレコード数を掛けた6レコードに増える。したがって、意味のあるレコードは網掛け部分の3行のみ。

FROM 句に複数のテーブルを並べたとき、SQL は自動的にお互いのテーブルの項目を掛け合わせて存在しないレコードまで作り出し、出力する行数が膨れあがってしまう。そのようなときに WHERE 仕入伝票表. 商品コード = 商品マスタ表. 商品コード のように、WHERE 句で実在するレコードに絞り込む。

この操作が、結合 (join) と選択である。

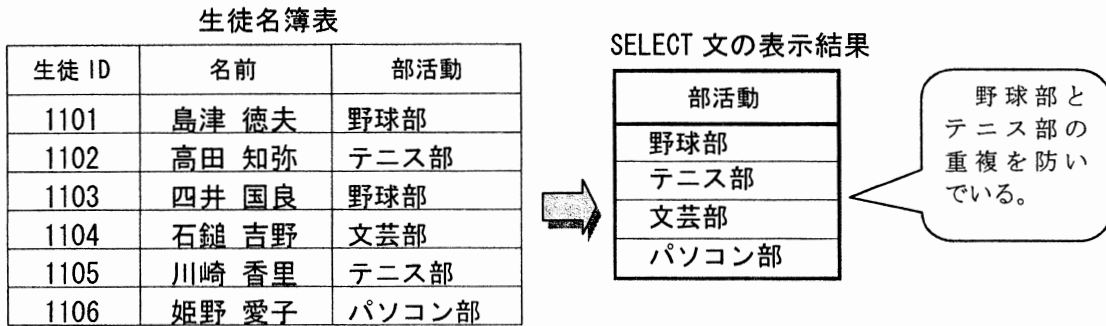
(3) SELECT 文の補足

ア SELECT DISTINCT ~

DISTINCT を SELECT の後に付けると、重複する内容は複数行ではなく 1 行だけで表示される。

例 SELECT DISTINCT 部活動 FROM 生徒名簿表

この SELECT 文で、生徒名簿表から重複なしで部活動を抜き出す。



2 集計関数

以下のように SQL の関数で自動的に集計計算できる。

| 機能 | 関数 | 使用例 |
|---------|-----------|--|
| 合計を答える | SUM(列名) | SELECT SUM(金額) FROM 入金伝票表 |
| 平均を答える | AVG(列名) | SELECT AVG(体重) FROM 身体計測表 |
| 最大値を答える | MAX(列名) | SELECT MAX(身長) FROM 身体計測表 |
| 最小値を答える | MIN(列名) | SELECT MIN(座高) FROM 身体計測表 |
| 件数を答える | COUNT(列名) | SELECT COUNT(名前) FROM 生徒表 WHERE 選択科目 = '音楽' |

例 SELECT SUM(得点) / COUNT(名前) AS 平均点 FROM 成績表

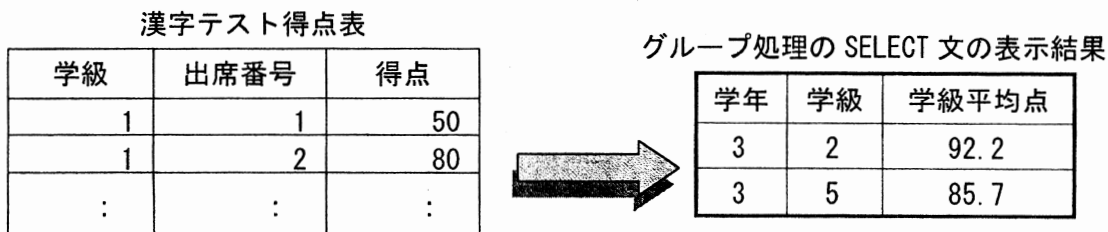
この SELECT 文で、成績表から平均点が計算され、平均点という別名 (エイリアス) で出力される。

3 GROUP BY 句の考え方と HAVING 句

SELECT 文に GROUP BY 句を付けると、特定の項目で集計して出力する。アンケートの集計や勘定科目ごとの金額合計などの処理に活用できる。

また、GROUP BY 句に、HAVIG を付けて、グループ化した行を絞り込むことができる。HAVING 句は、WHERE 句と違って、集計済みの表を絞り込む動作を行う。(例 WHERE 句で集計レコードを 3 年生だけに絞り込み、HAVING 句で表示する集計学級を平均点 80 点以上の学級のものに絞り込む)

例 SELECT 学年, 学級, AVG(得点) AS 学級平均点 FROM 漢字テスト得点表 WHERE 学年 = 3
GROUP BY 学級 HAVING 学級平均点 >= 80



※ ちょうど表計算ソフトのピボットテーブル機能と同様である。

4 WHERE 句の1級レベルの書き方

SELECT 文の WHERE 句の指定には、より細かい指定がある。

(1) LIKE 'OO' 指定

OO みたいな文字列を含むデータを選択する。

例 SELECT 商品コード, 商品 FROM 商品マスタ表 WHERE 商品 LIKE '新型%'

この SELECT 文で、商品マスタ表から「新型～」といった商品の商品コードと商品名を抽出する。

| 仕入伝票表 | | SELECT 文の表示結果 | |
|-------|--------|---------------|--------|
| 商品コード | 商品 | 商品コード | 商品 |
| 1 | 置き時計 | | |
| 2 | 新型置き時計 | 2 | 新型置き時計 |
| 3 | アイロン | | |
| 4 | スタンド | | |
| 5 | 新型スタンド | 5 | 新型スタンド |

また、LIKE 句で指定する文字列にあいまいな部分があるとき、ワイルドカードの記号を活用する。

ア %

LIKE 句の後の文字列に、% を付けると複数の任意の文字列を指す。

例 LIKE '新型%' 新型で始まる文字列を持つレコードを指定

LIKE '%プレーヤー' プレーヤーで終わる文字列を持つレコードを指定

LIKE '%青春%' 青春を含む文字列を持つレコードを指定

イ _

LIKE の後の文字列に、_ をつけると任意の1文字分を指す。

例 LIKE '山本_' 山本○ という文字列を持つレコードを指定。

LIKE '_町%' ○町～ という文字列を持つレコードを指定。

以上のように % と _ を組み合わせて、きめ細かい区別を指定した文字列探しを行う。

※ なお、ACCESS での操作では、ワイルドカードの記号は、%の代わりに?を、%の代わりには*を使う。

(2) BETWEEN ○ AND ○ 指定

○以上かつ○以下の範囲のデータを選択する。

例 SELECT 日付, 商品コード, 仕入数量 FROM 仕入伝票表 WHERE 仕入数量 BETWEEN 50 AND 100

この SELECT 文で、仕入伝票表から仕入数量 50 個以上 100 個以下のデータを選択し、その日付と商品コードと仕入数量を出力する。WHERE 仕入数量 >= 50 AND 仕入数量 <= 100 と同じ働きをする。

(3) IN (○, ○, …) 指定

○か、○か、… のどれかに一致するデータを選択する。

例 SELECT 日付, 勘定科目, 金額 FROM 入金伝票表

WHERE 勘定科目 IN ('売掛金', '貸付金', '受取手形')

この SELECT 文で入金伝票表から勘定科目が売掛金か貸付金か受取手形のデータを選択し、その日付と勘定科目と金額を抽出する。

WHERE 勘定科目 = '売掛金' OR 勘定科目 = '貸付金' OR 勘定科目 = '受取手形' と同じ働きをする。

5 副問い合わせ

仮想表（クエリ）を表（テーブル）のように扱って、FROM 句や WHERE 句に抽出する。

例 SELECT 日付, 商品, 仕入数量 FROM 仕入伝票表 WHERE 仕入先
IN (SELECT 仕入先 FROM 仕入先マスタ表 WHERE 住所 LIKE '愛媛県%')

この SELECT 文で、仕入伝票表から住所が愛媛県で始まる仕入先から仕入れたデータを選択し、その日付と仕入数量と仕入先を抽出する。なお、この住所が愛媛県から始まる仕入先の名前は、仕入先マスタ表で抽出（副問い合わせ）し、その抽出した仕入先名の並びを、IN()句で使っている。

副問い合わせの動作の例

| 通し番号 | 日付 | 仕入先コード | 商品 | 数量 |
|------|----------|--------|-------|----|
| 1 | 20080401 | 3 | デジカメ | 2 |
| 2 | 20080408 | 2 | 液晶テレビ | 4 |
| 3 | 20080412 | 3 | 液晶テレビ | 7 |
| 4 | 20080414 | 1 | デジカメ | 8 |
| 5 | 20080427 | 3 | 液晶テレビ | 6 |

SELECT 文の表示結果

| 日付 | 商品 | 仕入数量 |
|----------|-------|------|
| 20080408 | 液晶テレビ | 4 |
| 20080414 | デジカメ | 8 |

仕入先マスタ表

| 仕入先コード | 仕入先 | 郵便番号 | 住所 |
|--------|-------|---------|---------|
| 1 | 国際電器 | 7980010 | 愛媛県松山市 |
| 2 | YH 商事 | 7940500 | 愛媛県八幡浜市 |
| 3 | テレホー | 7660080 | 香川県高松市 |

副問い合わせが作る仮想表
(SELECT 仕入先 FROM 仕入先マスタ表
WHERE 住所 LIKE '愛媛県%')

| 仕入先 |
|-------|
| 国際電器 |
| YH 商事 |

この副問い合わせで、愛媛県にある仕入先データが求められ、国際電器と YH 商事が選ばれる。そして、元の SELECT 文の WHERE 句に、仕入先 IN ('国際電器', 'YH 商事') という完成した選択条件が埋め込まれ、表示結果のように抽出する。

6 SELECT 命令以外のデータベース操作命令(DML)

(1) UPDATE ~ SET ~ WHERE ~

指定した条件のレコードの項目を書き直す命令。

例 UPDATE 検定成績表 SET 合格 = 1 WHERE 筆記得点 >= 80 AND 実技得点 >= 80

この UPDATE 文によって、検定成績表で筆記得点も実技得点も 80 点以上のレコードを選び、合格フィールドに 1 を埋め込む。

(2) INSERT INTO ~ VALUES ~

テーブルに新しいレコードを追加する命令。

SQL データベースでは、レコードの追加位置は末尾だが、順番がそろっていないなくても困らない。なぜならば、項目にキーを設定することで目次情報が自動的に備わり、SELECT 文ですぐにその項目のデータの順番に並んだ状態で出力できるからである。

例 INSERT INTO 名簿表 (氏名, よみがな, 性別コード)
VALUES ('坂本 龍馬', 'さかもと りょうま', 1)

この INSERT 文で名簿表に新しく坂本龍馬のレコードが追加される。

(3) DELETE FROM ~ WHERE ~

テーブルから条件に合ったレコードを消去する命令。

WHERE 句を付けなければ全レコードを消す。

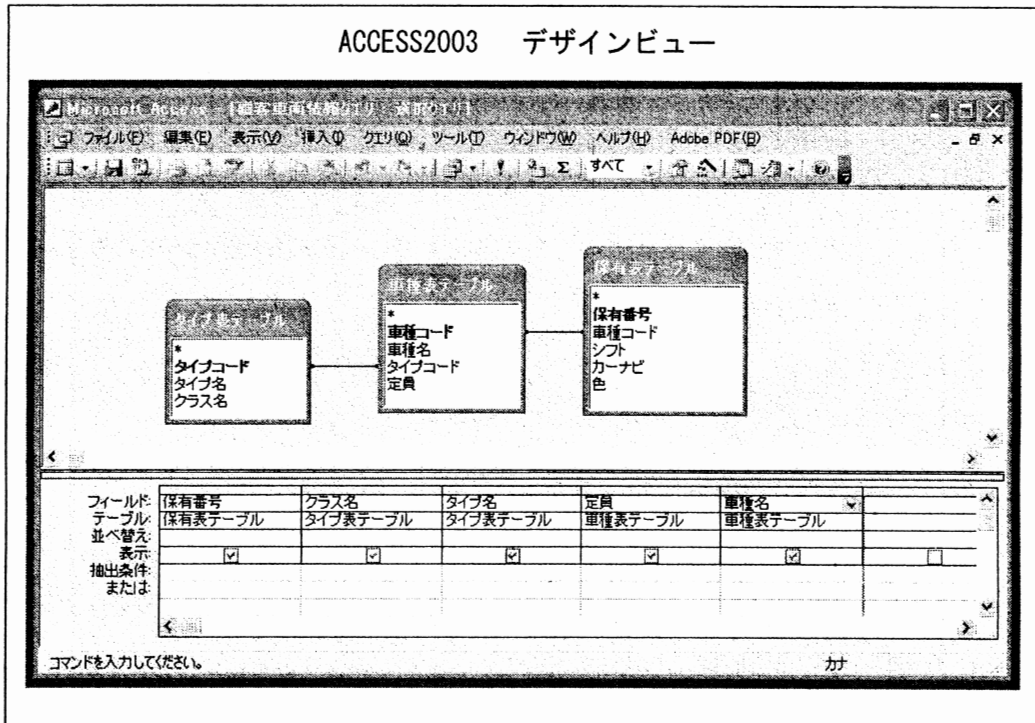
例 DELETE FROM 仕入伝票表 WHERE 日付 <= 20080331

この DELETE 文で、日付が 20080331 以下のレコードを消去する。

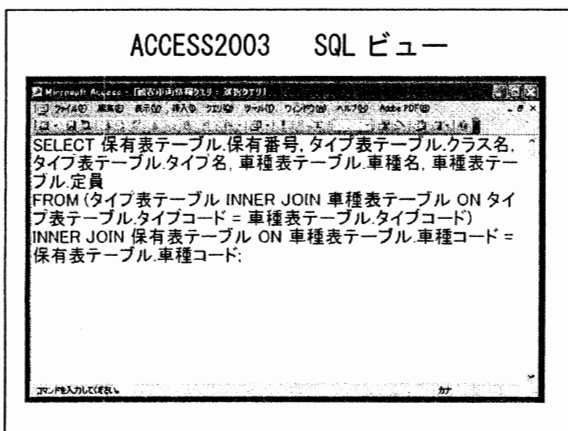
7 ACCESS データベースの SQL 機能

よく普及している MICROSOFT 製のリレーショナルデータベースソフト ACCESS を使って、自分で作ったデータベースの表を SQL 命令で操作することができる。

クエリを新規作成か開くかしてアクティブにする。そして、ツールバーの左上隅にある表示モードアイコンを SQL ビューにして、その枠内に正しい SQL 文を記述すれば、データシートビューにして動作を確認できる。クエリの手直しや SQL 文の動作テストに便利である。応用的な使い方として、解説 Web ページからの SQL 文の取り込んだり、一度、エディタにコピーして、置き換えや繰り返しコピーなどをしてから SQL ビューに貼り付けたりする使い方もできる。



顧客車両情報クエリの表示結果



| 保有番号 | クラス名 | タイプ名 | 車種名 | 定員 |
|------|---------|------|---------|----|
| H01 | ステーション | ワゴン | ウイングドーロ | 5 |
| H10 | ワンボックス | ワゴン | エレナ | 8 |
| H14 | ワンボックス | ワゴン | エレナ | 8 |
| H04 | ラグジュアリー | 乗用車 | ジーマ | 5 |
| H13 | ラグジュアリー | 乗用車 | ジーマ | 5 |
| H03 | ベーシック | 乗用車 | フローラ | 5 |
| H09 | ベーシック | 乗用車 | フローラ | 5 |
| H15 | ベーシック | 乗用車 | フローラ | 5 |
| H02 | ラグジュアリー | 乗用車 | マークⅢ | 5 |
| H05 | ラグジュアリー | 乗用車 | マークⅢ | 5 |
| H12 | ラグジュアリー | 乗用車 | マークⅢ | 5 |
| H06 | ベーシック | 乗用車 | ミニラ | 4 |
| H07 | ワンボックス | ワゴン | ローエース | 10 |
| H08 | ステーション | ワゴン | ワゴンでR | 4 |

計算問題の出題例および解答

〈平均故障間隔・平均修復時間・稼働率〉

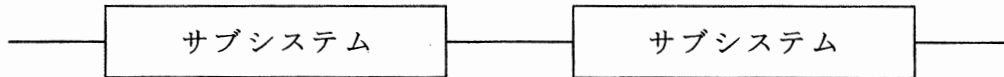
- 問 1 あるコンピュータシステムにおいて平均故障間隔と平均修復時間がともに 2 倍になったとき、このコンピュータの稼働率は何倍になるか。

1 倍

稼働率 = $MTBF \div (MTBF + MTTR)$ で求められるので

稼働率 = $MTBF \times 2 \div (MTBF \times 2 + MTTR \times 2)$ で稼働率は変わらない。

- 問 2 コンピュータシステムで、下図のように二つのサブシステムを接続した場合、このコンピュータシステム全体の稼働率を求めなさい。なお、各サブシステムの稼働率は 0.9 とする。



0.81

直列に接続した場合は、

0.9 (左のサブシステムの稼働率) × 0.9 (右のサブシステムの稼働率) となる。

〈転送時間・伝送時間・通信速度〉

- 問 3 10MB のデータをダウンロードするのに 2 秒かかった。この回線の伝送効率 50% としたとき、この回線の通信速度を計算しなさい。

20Mbps

1 秒間あたりの通信できるデータ量は、 $10MB \div 2 \text{ 秒} \div 0.5$ (伝送効率) = 2.5MB

1 バイトを 1 ビットになおして、 $2.5MB \times 8 = 20Mbps$

〈MIPS〉

- 問 4 30MIPS の CPU が 300 万回の命令を実行するのに何秒かかるか求めなさい。

0.1 秒

1MIPS の CPU は 1 秒間に 100 万回の命令が実行できるから、

$3,000,000 \div 30,000,000 = 0.1$

〈記憶容量〉

- 問 5 過去の検定問題の内容を CD-R に記録する場合、1 枚の CD-R におよそ何回分記録できるか。CD-R の記録容量は 700MB で、1 回の検定問題は合計で 7 ページ 1 ページの容量は 10MB とする。それを圧縮率 50% で記録できるものとする。

20 回分

1 回の検定問題あたりの記憶容量

$10MB$ (1 ページの容量) × 7 (ページ) × 50% (圧縮率) = 35MB

$700MB \div 35MB = 20$ (回分)

〈画像容量〉

- 問 6 解像度 1,200 × 1,000 ドットで、24 ビットの色情報をもつ画像で撮影できるデジタルカメラがある。このカメラに 20 枚の画像を保存するのに適した記録用メモリを選びなさい。なお、すべて同じ条件で撮影し、データは圧縮しないものとする。ただし、保存できる最低の容量をもつ記録用メモリを選択すること。

ア. 64 MB

イ. 128 MB

ウ. 256 MB

この画像の 1 枚あたりのドット数は、 $1,200 \times 1,000 = 1,200,000$ ドット

この画像の 1 枚あたりの容量は

$1,200,000 \times (24 \text{ ビット} \div 8 \text{ ビット}) = 3,600,000$ バイト (3.6MB)

20 枚保存したいから、 $3.6MB \times 20 \text{ 枚} = 72MB$

問 1

次の表は、ある検定試験の受験料検索表である。B5の「受験料」は、B3の「検定名」とB4「級」をもとに、受験料一覧表を参照して表示する。B5に設定する式として適切なものを選び、記号で答えなさい。

| | A | B | C | D | E |
|----|--------|-------|-------|-------|-------|
| 1 | | | | | |
| 2 | 受験料検索表 | | | | |
| 3 | 検定名 | 情報処理 | | | |
| 4 | 級 | 2 | | | |
| 5 | 受験料 | 1,200 | | | |
| 6 | | | | | |
| 7 | 受験料一覧表 | | | | |
| 8 | | 検定名 | | | |
| 9 | 級 | 簿記 | 情報処理 | ワープロ | 英語 |
| 10 | 1 | 2,000 | 1,500 | 1,600 | 1,000 |
| 11 | 2 | 1,000 | 1,200 | 1,400 | 1,000 |
| 12 | 3 | 1,000 | 000 | 1,200 | 900 |

関数の引数

INDEX

配列 B10:E12 = {2000,1500,1600,100}

行番号 B4 = 2

列番号 MATCH(B3,B9:E9,0) = 2

= 1200

インデックスを使って、範囲、または配列から抽出した値を返します。

列番号には <配列> の中にある、値を返す列を指定します。

数式の結果 = 1,200

[この関数のヘルプ\(H\)](#) OK キャンセル

- ア =INDEX (B10:E12,B4,MATCH (B3,B9:E9,0))
- イ =INDEX (B10:E12,MATCH (B3,B9:E9,0) B4)
- ウ =VLOOKUP (B4,A1-0:E12,MATCH (B3,B9:E9,0),FALSE)

MATCH 関数は、検定名 (B9 ~ E9) の中から「情報処理」を検索し、左端からの位置を求める。

問 2

ある商店では、次の表を用いて、商品をケース単位で注文している。C 4 「仕入本数」は、C 4 の「仕入本数」は、B 4 の「販売予想数」以下で、A 4 の「1ケースの本数」の倍数で、最も大きい数値を求める。C 4 に設定する式として適切なものを選び、記号で答えなさい。

| | A | B | C |
|---|---------|-------|------|
| 1 | | | |
| 2 | 仕入本数計算表 | | |
| 3 | 1ケースの本数 | 販売予想数 | 仕入本数 |
| 4 | 24 | 172 | 168 |

関数の引数

FLOOR

数値 B4 = 172

基準値 A4 = 24

= 168

指定した数値よりも 0 に近い数値に丸めた値を返します。

基準値 には計算の最小単位となる数値を指定します。

数式の結果 = 168

[この関数のヘルプ\(H\)](#) OK キャンセル

- ア $=\text{FLOOR}(B4,A4) \rightarrow 168$ 172 を 24 の倍数になるように切り捨てた数値を求める。
- イ $=\text{CELLING}(B4,A4) \rightarrow 192$ 172 を 24 の倍数になるように切り上げた数値を求める。
- ウ $=\text{MOD}(B4,A4) \rightarrow 4$ 172 ÷ 24 の余りを求める。

問 3

ある会社では、次の表を用いて、社員の勤務開始時間を計算している。C 4 の「勤務開始時間」は、B 4 の「出勤時間」を 15 分単位で切り上げて求める。C 4 に設定する式として適切なものを選び、記号で答えなさい。

| | A | B | C |
|---|-----------|------|--------|
| 1 | | | |
| 2 | 勤務開始時間計算表 | | |
| 3 | 社員名 | 出勤時間 | 勤務開始時間 |
| 4 | 松坂〇〇 | 8:10 | 8:15 |
| 5 | 岡島〇〇 | 8:24 | 8:30 |
| 6 | 松井〇〇 | 8:30 | 8:30 |
| 7 | 井口〇〇 | 8:42 | 8:45 |

関数の引数

CEILING

数値 B4 = 0.340277778

基準値 TIME(0,15,0) = 0.010416667

= 0.34375

指定した数値をはさむ基準値の倍数のうち、0 から遠い方の値を返します。

数値 には数値を指定します。

数式の結果 = 8:15

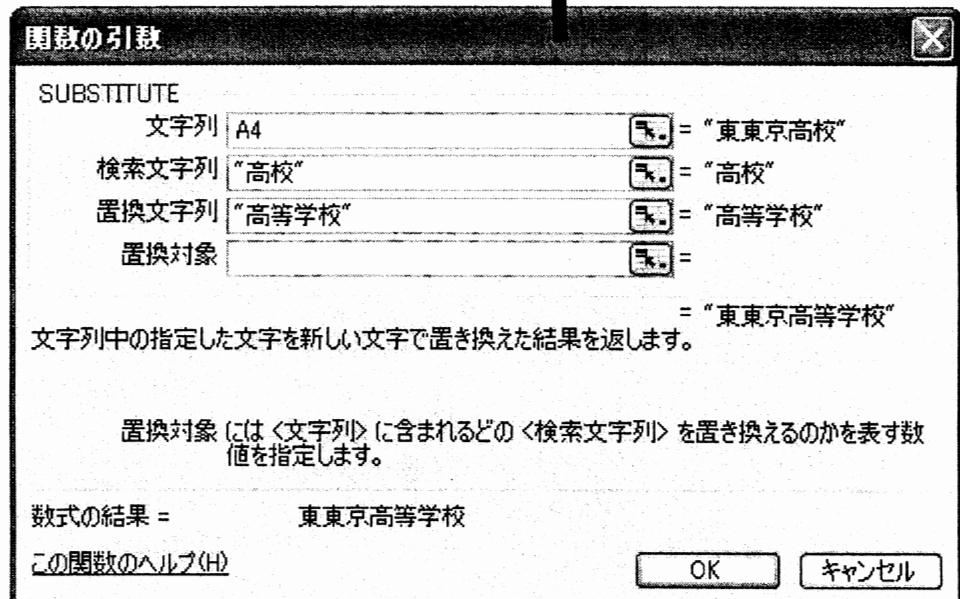
[この関数のヘルプ\(H\)](#) OK キャンセル

- ア $=\text{CEILING}(B4,\text{TIME}(0,15,0)) \rightarrow 8:15$
 8:10 を 15 分の倍数になるように切り上げた数値を求める。
- イ $=\text{FLOOR}(B4,\text{TIME}(0,15,0)) \rightarrow 8:00$
 8:10 を 15 分の倍数になるように切り捨てた数値を求める。
- ウ $=\text{INT}(B4/\text{TIME}(0,15,0)) \rightarrow 32$
 8:10 を 15 分で割り、切り捨てた整数を求める。

問 4

次の表は、文字を検索して置き換える表である。B 4 の「変更後」は、A 4 の「変更前」の文字列に高校がある場合は高等学校に置き換え、それ以外の場合は「変更前」の文字を表示する。B 4 に設定する式として適切なものを選び、記号で答えなさい。

| | A | B |
|---|---------|-----------|
| 1 | | |
| 2 | 検索置換表 | |
| 3 | 変更前 | 変更後 |
| 4 | 東東京高校 | 東東京高等学校 |
| 5 | 西神奈川小学校 | 西神奈川小学校 |
| 6 | 南千葉中学校 | 南千葉中学校 |
| 7 | 北埼玉商業高校 | 北埼玉商業高等学校 |

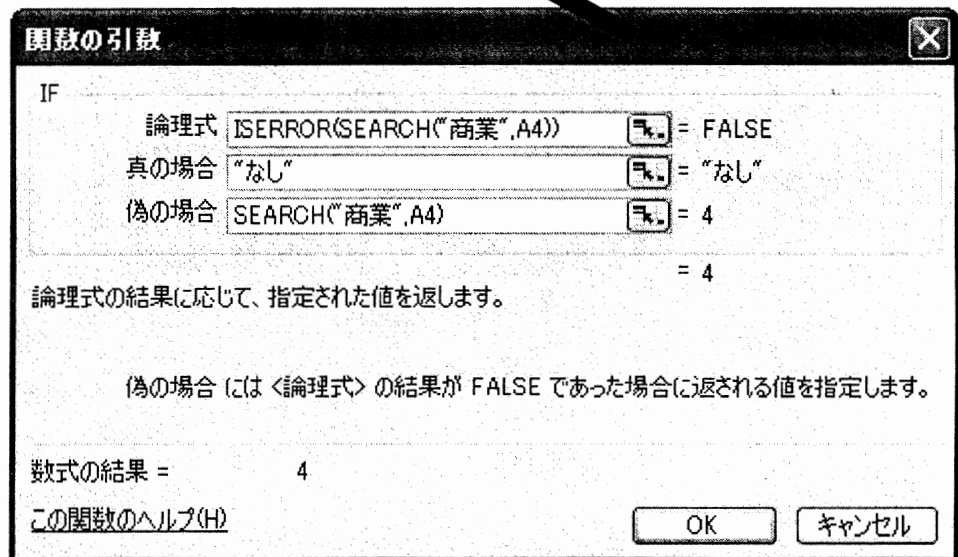


- ア $=\text{SUBSTITUTE}(A4,"高校","高等学校")$
 SUBSTITUTE 関数は、「高校」を「高等学校」に置き換える。
- イ $=\text{SEARCH}("高校",A4,"高等学校")$
 SEARCH 関数は、検索文字列から、対象となる文字の開始位置を検索する。
- ウ $=\text{IF}(A4="",A4&"高等学校",A4)$
 A4 のセルが空白であれば、「高等学校」を表示し、そうでなければ A 4 のセル内容を表示する。

問 5

次の表は、文字を検索する表である。B 4 の「検索結果」は、A 4 の「学校名」の文字列に商業がある場合左端からの文字位置を求め、それ以外の場合はなしを表示する。B 4 に設定する式として適切なものを選び、記号で答えなさい。

| | A | B |
|---|----------|------|
| 1 | | |
| 2 | 学校名検索表 | |
| 3 | 学校名 | 検索結果 |
| 4 | 東東京商業高校 | 4 |
| 5 | 西神奈川商業高校 | 5 |
| 6 | 南千葉農業高校 | なし |
| 7 | 北埼玉商業高校 | 4 |



- ア =IF (ISERROR (SEARCH ("商業",A4)), "なし",SEARCH ("商業",A4))
- イ =IF (ISERROR (SEARCH ("商業",A4),SEARCH ("商業",A4),"なし")
- ウ =IF (SEARCH ("商業",A4)>=1,SEARCH ("商業",A4),"なし")

SEARCH 関数は、「東東京商業高校」から「商業」の文字位置を決める。
ISERROR 関数は、SEARCH ("商業",A4) がエラーの場合は TRUE を返す。