

問1 データベースの物理設計とデータ移行に関する次の記述を読んで、設問1~3に答えよ。

D銀行は、首都圏の100支店に、投資信託などの運用商品を扱う部課があり、専任の営業員が個人顧客（以下、顧客という）への販売活動を行っている。営業員の販売活動には、顧客の個人情報、資産情報、投資経験、取引などを格納した顧客情報管理システムが利用されている。D銀行では、顧客情報管理システムのDBサーバ及びストレージの老朽化に伴うリプレース（以下、DBサーバ更改という）に当たって、RDBMSのバージョンアップ、アプリケーション（以下、APという）の機能追加などを行うことにした。

#### [データベースの論理設計]

顧客情報管理システムの概念データモデルを図1に、テーブル構造を図2に、主な列とその意味・制約を表1に示す。

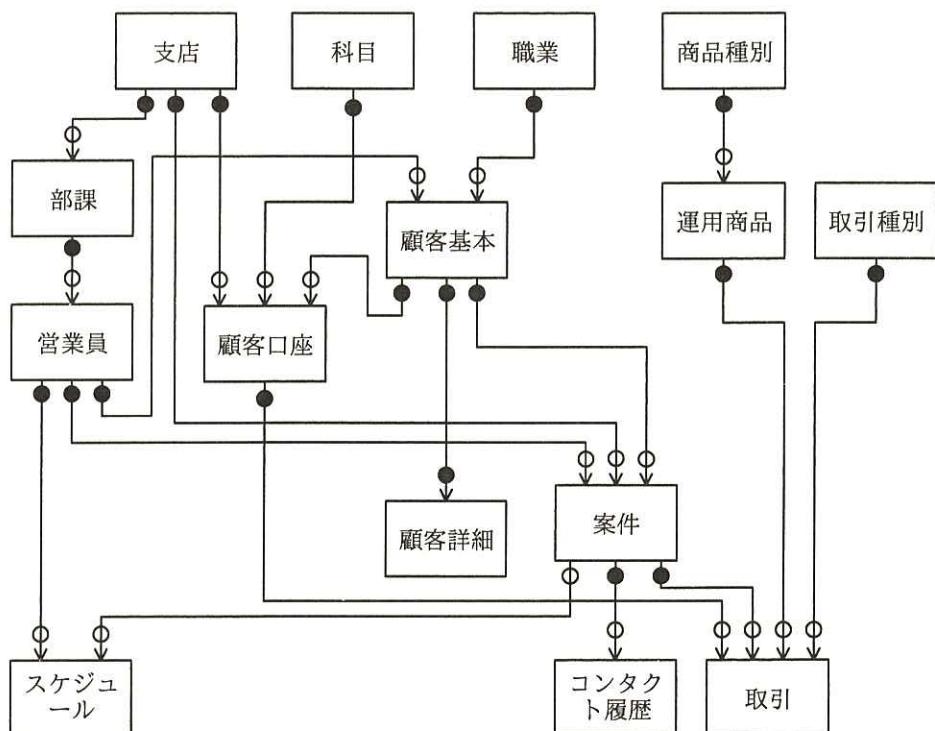


図1 概念データモデル

支店（支店コード, 支店名, 最終更新 TS）  
部課（部課コード, 支店コード, 部課名, 最終更新 TS）  
営業員（行員番号, 行員名, 部課コード, 最終更新 TS）  
科目（科目コード, 科目名, 最終更新 TS）  
職業（職業コード, 職業名, 最終更新 TS）  
取引種別（取引種別コード, 取引種別名, 最終更新 TS）  
商品種別（商品種別コード, 商品種別名, 最終更新 TS）  
運用商品（商品コード, 商品種別コード, 商品名, 商品説明, 最終更新 TS）  
顧客基本（顧客番号, 顧客名, 住所, 電話番号, 生年月日, 性別, 職業コード, 勤務先,  
担当行員番号, 最終更新 TS）  
顧客詳細（顧客番号, 更新年月日, 家族構成, 資産総額, 投資可能額, 投資信託経験年数,  
外貨預金経験年数, …）  
顧客口座（支店コード, 科目コード, 口座番号, 顧客番号, 開設日, 期首残高, 現在残高,  
最終更新 TS, …）  
案件（支店コード, 案件番号, 行員番号, 顧客番号, 登録日, 状態, 最終更新 TS, …）  
取引（支店コード, 案件番号, 取引番号, 最終更新 TS, 取引種別コード, 商品コード,  
取引日, 科目コード, 口座番号, 取引口数, 取引金額, …）  
コンタクト履歴（支店コード, 案件番号, コンタクト日時, 最終更新 TS, 会話 1,  
会話 2, …）  
スケジュール（行員番号, 予定日, 開始時刻, 行番号, 終了時刻, 行動種別, 行動内容,  
支店コード, 案件番号）

図 2 テーブル構造（一部省略）

表1 主な列とその意味・制約

列名	意味・制約
支店コード	支店を一意に識別するコード（4桁の半角英数字）
顧客番号	銀行内で顧客を一意に識別する番号（1～999,999,999）
顧客名	顧客の氏名（全角文字30字以内。平均文字数は5文字）。“顧客基本”テーブルの顧客名は、指定した文字列との完全一致、部分一致（あいまい検索）などの条件検索に使用される。
住所	顧客の住所（全角文字30字以内。平均文字数は19文字）
性別	‘0’（未記入）、‘1’（男性）、‘2’（女性）のいずれか
行員番号	銀行内で役員・行員を一意に識別する番号（1,000,001～9,999,999）
科目コード	普通預金、定期預金、投資信託などの種別を識別するコード（3桁の半角英数字）
口座番号	顧客の口座を支店コードごと、科目コードごとに一意に識別する番号（7桁の半角英数字）
期首残高、現在残高	残高は、顧客の各口座における、ある時点の保有金額である。期首残高は期首日（4月1日）時点の残高、現在残高は最新の残高を表す。
案件番号	支店ごとに案件を一意に識別する番号（1～99,999,999）。“スケジュール”テーブルの支店コードと案件番号は、案件に関連する顧客との面談予定などを登録する場合にだけ設定する。
登録日	案件を最初に起案した年月日。“案件”テーブルの場合は必須で、変更することはない。
予定日	会議、顧客往訪などの行動が予定されている年月日
開始時刻、終了時刻、行番号	開始時刻、終了時刻は、スケジュールの開始時分と終了時分。“スケジュール”テーブルの場合は必須で、終了時刻は開始時刻よりも後でなければならない。行番号は、同じ開始時刻で複数の予定を登録できるように、1～999の連番で区別する。
行動種別	‘1’（会議）、‘2’（打合せ）、‘3’（顧客往訪）、‘4’（顧客来訪）、‘5’（作業）のいずれか
行動内容	具体的な行動内容（全角文字1,000字以内。平均文字数は58文字）。“スケジュール”テーブルの場合は任意である。
最終更新 TS	テーブルの行が、追加又は最後に更新された時刻印（年月日時分秒）。システムで自動設定する。

#### 〔顧客情報管理システムの現状とDBサーバ更改〕

##### 1. 現状

- (1) 顧客情報管理システムは、Webサーバ、APサーバ、DBサーバから成るシステムで、内部ネットワーク上に設置されている。内部ネットワークは、外部ネットワークから隔離されているので、外部から侵入されることはない。
- (2) 案件、取引、コンタクト履歴には、案件の登録日が10年前の月初めから現在までの全てのデータが保存されている。コンタクト履歴は、過去のデータが更新されることもある。保存期間が過ぎた案件及び案件に関連するデータは、毎月末日のバッチ処理で削除される。

##### 2. DBサーバ更改の方針

- (1) 新 DB サーバを導入し、現行 DB サーバと同じネットワーク上に設置する。
- (2) 現行 DB サーバのデータを新 DB サーバに移行し、新 DB のデータ検証後に AP サーバからの接続先を新 DB サーバに切り替える。
- (3) 本番開始前 1 か月程度の移行準備期間（以下、準備期間という）には、日々のバッチ処理終了後の 6 時間だけサービスを停止する。本番開始直前のデータ移行・検証期間（以下、移行期間という）は、連続して 2 日間サービスを停止する。本番開始後には、1 か月程度の特別な監視体制の維持期間（以下、監視期間という）を設ける。
- (4) データ移行時には、その月の末日のバッチ処理で削除されることが分かっているデータは、移行の対象としない。また、各テーブルの最終更新 TS の列値は、システムで自動設定せず、そのまま移行する。

### 3. DB サーバ更改時の実施事項

- (1) 現行 DB サーバで稼働する RDBMS は、上位バージョンに移行する。
- (2) 顧客とのアポイントメントを案件と関係付けて管理するために、スケジュール管理機能を追加する。スケジュール管理機能の提供は、DB サーバ更改に合わせて開始する。
- (3) 個人情報の漏えい対策を強化する。
- (4) 取扱商品の増加に備えて、商品コードの桁数を 4 桁から 6 桁に増やす。移行時に、現在の商品コードの前に ‘00’ を付加する。

### 〔RDBMS の仕様〕

#### 1. 表領域

- (1) テーブル、索引などのストレージ上の物理的な格納場所を、表領域という。
- (2) RDBMS とストレージ間のデータ入出力単位を、データページという。データページには、テーブル、索引のデータが格納される。表領域ごとに、ページサイズ（1 データページの長さ。2,000, 4,000, 8,000, 16,000 バイトのいずれかである）と、空き領域率（将来の更新に備えて、データページ内に確保しておく空き領域の割合）を指定する。
- (3) 同じデータページに、異なるテーブルの行が格納されることはない。

#### 2. テーブル

- (1) テーブルの列には、NOT NULL 制約を指定することができる。NOT NULL 制約を指定しない列には、NULL かどうかを表す 1 バイトのフラグが付加される。
- (2) 主キー制約には、主キーを構成する列名を指定する。
- (3) 参照制約には、列名、参照先テーブル名、参照先列名を指定する。
- (4) 検査制約には、同一行の列に対する制約を指定する。
- (5) 使用可能なデータ型は、表 2 のとおりである。

表 2 使用可能なデータ型

データ型	説明
CHAR(n)	n 文字の半角固定長文字列 ( $1 \leq n \leq 255$ )。文字列が n 字未満の場合は、文字列の後に半角の空白を挿入し、n バイトの領域に格納される。
NCHAR(n)	n 文字の全角固定長文字列 ( $1 \leq n \leq 127$ )。文字列が n 字未満の場合は、文字列の後に全角の空白を挿入し、“n×2” バイトの領域に格納される。
VARCHAR(n)	最大 n 文字の半角可変長文字列 ( $1 \leq n \leq 8,000$ )。“文字列の文字数” バイトの領域に格納され、4 バイトの制御情報が付加される。
NCHAR VARYING(n)	最大 n 文字の全角可変長文字列 ( $1 \leq n \leq 4,000$ )。“文字列の文字数×2” バイトの領域に格納され、4 バイトの制御情報が付加される。
SMALLINT	-32,768 ~ 32,767 の範囲内の整数。2 バイトの領域に格納される。
INTEGER	-2,147,483,648 ~ 2,147,483,647 の範囲内の整数。4 バイトの領域に格納される。
DECIMAL(m,n)	精度 m ( $1 \leq m \leq 31$ )、位取り n ( $0 \leq n \leq m$ ) の 10 進数。“m÷2+1” の小数部を切り捨てたバイト数の領域に格納される。
DATE	0001-01-01 ~ 9999-12-31 の範囲内の日付。4 バイトの領域に格納される。
TIME	00:00:00 ~ 23:59:59 の範囲内の時刻。3 バイトの領域に格納される。
TIMESTAMP	0001-01-01 00:00:00.000000 ~ 9999-12-31 23:59:59.999999 の範囲内の時刻印。10 バイトの領域に格納される。

### 3. 索引

索引には、ユニーク索引と非ユニーク索引がある。

### 4. DB の相互接続機能

同じネットワーク上にある異なる DB を、相互接続する機能である。相互接続した DB 上のテーブルは、あたかも同じ DB 上にあるかのように、一つの SQL 文で操作することができる。SQL 文では、テーブル名を DB の識別子とスキーマ名の両方で修飾することができる。

### 5. エクスポートツール、インポートツール

テーブルのデータをテキストファイルに出力するエクスポートツールと、テキストファイルのデータをテーブルに入力するインポートツールがある。エクスポートツールでは、SQL の SELECT 文を指定することができる。一方、インポートツールでは、SQL の INSERT 文と同じ処理が行われる。

#### [データベースの物理設計]

##### 1. テーブル定義

次の方針に基づいて、テーブル定義表を作成し、テーブル定義を行う。

- (1) データ型欄には、データ型、データ型の適切な長さ、精度、位取りを記入する。データ型の選択は、次の規則に従う。
  - ① 文字列型の列が全角文字の場合は、NCHAR 又は NCHAR VARYING を選択し、それ以外の場合は CHAR 又は VARCHAR を選択する。
  - ② 数値の列が整数である場合は、取り得る値の範囲に応じて、SMALLINT 又は INTEGER を選択する。それ以外の場合は DECIMAL を選択する。
  - ③ ①及び②どちらの場合も、列の取り得る値の範囲に従って、格納領域の長さが最小になるようにデータ型を選択する。
  - ④ 日付、時刻、時刻印の列は、専用のデータ型を選択する。
- (2) NOT NULL 欄には、NOT NULL 制約がある場合は Y を、ない場合は N を記入する。
- (3) 格納長欄には、RDBMS の仕様に従って、格納長を記入する。可変長文字列の格納長は、表 1 から平均文字数が分かる場合はそれを基準に算出し、それ以外の場合は最大文字数の半分を基準に算出する。
- (4) 索引の種類と構成列欄には、作成する索引を記入する。
  - ① 索引の種類には、P (主キーの索引), U (ユニーク索引), NU (非ユニーク索引) がある。
  - ② 主キーの索引は、必ず作成する。
  - ③ 主キー以外で値が一意となる列又は列の組合せには、必ずユニーク索引を作成する。それ以外の列又は列の組合せが、外部キーを構成する場合は、必ず非ユニーク索引を作成する。
  - ④ 各索引の構成列には、構成列の定義順に 1 からの連番を記入する。

- (5) 制約欄には、参照制約、検査制約を SQL の構文で記入する。  
 “顧客基本”，“スケジュール” テーブルのテーブル定義表を表 3, 表 4 に示す。

表 3 “顧客基本” テーブルのテーブル定義表

項目 列名	データ型	NOT NULL	格納長 (バイト)	索引の種類と構成列			
				P	NU	NU	NU
顧客番号	INTEGER	Y	4	1			
顧客名	NCHAR VARYING(30)	Y	14		1		
住所	NCHAR VARYING(200)	Y	42				
電話番号	VARCHAR(20)	Y	14				
生年月日	DATE	Y	4				
性別	CHAR(1)	Y	1				
職業コード	CHAR(4)	Y	4			1	
勤務先	NCHAR VARYING(100)	N	105				
担当行員番号	INTEGER	N	5				1
最終更新 TS	TIMESTAMP	Y	10				
制約	FOREIGN KEY (職業コード) REFERENCES 職業 (職業コード) FOREIGN KEY (担当行員番号) REFERENCES 営業員 (行員番号) CHECK (性別 IN ('0', '1', '2'))						

表 4 “スケジュール” テーブルのテーブル定義表（作成中）

項目 列名	データ型	NOT NULL	格納長 (バイト)	索引の種類と構成列		
行員番号	INTEGER	Y	4			
予定日	DATE	Y	4			
開始時刻						
行番号						
終了時刻						
行動種別						
行動内容						
支店コード						
案件番号						
制約	FOREIGN KEY (行員番号) REFERENCES 営業員 (行員番号) FOREIGN KEY [a] CHECK ([b]) CHECK ([c])					

## 2. データ所要量の見積り

“スケジュール” テーブルのデータ所要量を見積もるために、表 5 を作成した。なお、データ所要量は、項番 1～5 の値を用いて算出するものとする。

表 5 “スケジュール” テーブルのデータ所要量（未完成）

項目番号	項目	値
1	見積行数	1,200,000 行
2	ページサイズ	4,000 バイト
3	平均行長	<input type="text"/> d バイト
4	1 データページ当たりの平均行数	<input type="text"/> e 行
5	必要データページ数	<input type="text"/> f ページ
6	データ所要量	<input type="text"/> g 百万バイト

## 3. 個人情報保護の強化

外部ネットワークから隔離されていても、内部の人間の犯行などによる個人情報の漏えいが懸念される。強化策として、次に示す“顧客基本” テーブル実装案を検討し、想定されるセキュリティ事故 A～C への対策の有効性を評価して、表 6 にまとめた。

案：顧客名、住所、電話番号の列値を、次のように格納、検索する。

- AP は、RDBMS の暗号化関数によってデータを暗号化し、SQL 文によって列に格納する。このとき、各列のデータ型、データ長の定義を変更する。
- AP は、SQL 文によって暗号化したデータを検索し、RDBMS の復号関数によって復号する。
- 暗号化と復号の鍵は、AP 内に埋め込み、外部から見えないようにする。

表 6 セキュリティ事故と対策の有効性評価（未完成）

記号	セキュリティ事故	有効性
A	営業員が顧客情報照会の AP を利用して、複数顧客の顧客名、住所、電話番号を検索して得られたデータを不正に持ち出した。	<input type="text"/> h
B	顧客情報管理システムの DB のフルバックアップを格納したメディアが盗難に遭った。	<input type="text"/> i
C	顧客情報管理システムの DB へのログイン ID とパスワードが盗まれ、AP を利用せずに、“顧客基本” テーブルに対する SELECT 文が不正に実行された。	△

注記 ○：有効である △：有効でない場合がある ×：有効でない

[データ移行の前提・移行計画・データ移行処理方式設計]

### 1. データ移行の前提

#### (1) テーブルへのデータ投入順序

テーブルへのデータ投入順序を、表 7 に示す。表 7 では、図 2 の“スケジュール”以外のテーブルを、参照制約違反が発生しない限り、できるだけ小さい枠番号のいずれかに配置し、枠番号順にデータ投入を行う。同じ枠番号内のテーブルへのデータ投入は、並行して行われる。

表 7 テーブルへのデータ投入順序（未完成）

枠番号	テーブル名
1	支店, 職業, 科目, 取引種別, j
2	部課, k
3	営業員
4	l
5	案件, m
6	n

#### (2) データ量の調査

現行 DB の主なテーブルの統計情報を、表 8 に示す。

表 8 現行 DB の主なテーブルの統計情報

テーブル名	平均行長 (バイト)	行数(行)	ページサイズ (バイト)	使用ページ数 (ページ)
商品種別	100	60	2,000	4
運用商品	600	1,300	4,000	217
顧客基本	203	5,100,000	4,000	300,000
顧客詳細	4,000	48,000,000	16,000	16,000,000
顧客口座	240	30,000,000	4,000	2,000,000
案件	1,500	72,000,000	8,000	18,000,000
取引	400	216,000,000	8,000	12,000,000
コンタクト履歴	1,200	576,000,000	8,000	96,000,000

#### (3) データ移行方法

現行 DB から新 DB にデータを移行するには、次に示す二つの方法がある。二

つ の方法について、それぞれ主なテーブルのデータ移行処理速度を実測し、表 9 を作成した。

① エクスポート・インポート (E&I)

現行 DB のテーブルからデータをエクスポート (EXP) したファイルを新 DB サーバに転送し、新 DB のテーブルにインポート (IMP) する。EXP と IMP には、それぞれエクスポートツールとインポートツールを用いる。

② ダイレクトインサート (DI)

DB の相互接続機能を利用して、SQL 文の実行によって、現行 DB のテーブルのデータを読み込み、新 DB のテーブルのデータを直接、追加、更新、削除する。

表 9 主なテーブルのデータ移行処理速度

テーブル名	EXP (行／秒)	IMP (行／秒)	DI (行／秒)
商品種別	12,000	4,000	3,000
顧客詳細	8,000	4,000	2,500
コンタクト履歴	10,000	5,000	2,000

注記 ファイル転送の処理時間は、無視できるものとする。

## 2. 移行計画

移行計画に先立ち、主要な移行工程・所要時間を表 10 にまとめた。

表 10 主要な移行工程・所要時間（未完成）

期間	工程	内容	所要時間
準備期間	①新 DB 準備	新 DB に DB オブジェクトを作成する。	5 日
	②データ移行	現行 DB のデータを新 DB に移行する。	(検討中)
移行期間	③差分データ反映	②の後で追加又は更新されたデータを、新 DB に反映する。	(検討中)
	④データ検証	現行 DB と新 DB のデータの整合性を検証する。	8 時間
	⑤バックアップ	新 DB のフルバックアップを取得する。	10 時間
	⑥判定	先行工程の結果を評価して、本番開始の可否を判定する。	2 時間
	⑦本番開始	AP サーバからの接続先を新 DB サーバに切り替える。	2 時間
監視期間	⑧本番監視	障害、性能、業務データの整合性を監視し、問題があれば対処する。	1か月

### 3. データ移行処理方式設計

表 10 の工程②データ移行について、データ量が最も多い“コンタクト履歴”テーブルについて、次のように検討した。

- (1) 表 8 と表 9 から、全行のデータ移行時間は、E&I では  時間、DI では、 p 時間である。準備期間内のバッチ処理後のサービス停止時間を利用して移行処理を毎日 4 時間ずつ行うと、最短でも  q 日掛かる。
- (2) 準備期間内のデータ移行処理では、エクスポートツールに指定する SQL 文を処理日別に用意すると、抽出条件の誤り、実行順の誤りなどが懸念される。そこで、移行データ抽出に使用する“抽出範囲”テーブルを現行 DB に追加し、抽出処理日ごとに、行数が均等になるように支店コードの範囲を決めて登録することにした。“抽出範囲”テーブルのテーブル構造を図 3 に、エクスポートツールに指定する SQL 文を図 4 に示す。

抽出範囲（抽出処理日、開始支店コード、終了支店コード）

図 3 “抽出範囲” テーブルのテーブル構造

SELECT A.\* FROM コンタクト履歴 A, 抽出範囲 B WHERE  r AND  s

図 4 エクスポートツールに指定する SQL 文（未完成）

- (3) 表 10 の工程③差分データ反映で行う二つの処理を表 11 にまとめた。

なお、処理 A と処理 B は、相互に現行 DB 上の異なる行を対象とする。

表 11 差分データ反映の処理内容とデータ移行方法（未完成）

処理	内容	データ移行方法
処理 A	“抽出範囲” テーブルの行の開始支店コードと終了支店コードの範囲内で、支店コードが一致する“案件” テーブルの行に対応し、かつ、次の条件に一致する“コンタクト履歴” テーブルの行を、新 DB に追加する。 条件：登録日が抽出処理日の後である	<input type="text"/> v
処理 B	“抽出範囲” テーブルの行の開始支店コードと終了支店コードの範囲内で、支店コードが一致する“案件” テーブルの行に対応し、かつ、次の条件に一致する“コンタクト履歴” テーブルの行の列値で新 DB を更新する。 条件： <input type="text"/> t かつ <input type="text"/> u	<input type="text"/> w

(データ移行後の検証)

DB の相互接続機能を利用して、現行 DB と新 DB のテーブルを比較し、SQL 文を実行して全行、全列の値一致を確認する。二つの集合が同一であることを確かめる論理式と、対応する SQL 文を、表 12 にまとめた。さらに、表 12 の SQL①を基に、“取引” テーブルを検証するための SQL 文を、図 5 に示す。

表 12 論理式と対応する SQL 文（未完成）

論理式	対応する SQL 文	
A-B = $\phi$ かつ $B-A = \phi$	SELECT COUNT(*) FROM (SELECT * FROM A EXCEPT SELECT * FROM B) R1	SQL①
	SELECT COUNT(*) FROM (SELECT * FROM B EXCEPT SELECT * FROM A) R2	SQL②
$(A \cup B) - (A \cap B) = \phi$	SELECT COUNT(*) FROM ((SELECT * FROM A [x] SELECT * FROM B) [y] (SELECT * FROM A [z] SELECT * FROM B)) R3	SQL③

凡例  $A \cup B$  : A と B の和集合  $A - B$  : A と B の差集合  $A \cap B$  : A と B の積集合  $\phi$  : 空集合

```
SELECT COUNT(*)
FROM ((SELECT * FROM OLD_DB.C01.取引)
      EXCEPT
      (SELECT * FROM NEW_DB.C01.取引)) R1
```

注記 OLD\_DB : 現行 DB の識別子、NEW\_DB : 新 DB の識別子、C01 : スキーマ名

図 5 “取引” テーブルを検証するための SQL 文

設問 1 【データベースの物理設計】について、(1)~(3)に答えよ。

- (1) 表 4 の太枠内に適切な字句を記入して太枠内を完成させよ。ただし、索引の種類と構成列の欄は全て埋まるとは限らない。

また、[a] ~ [c] に入る適切な字句を答えよ。ただし、1 ~999 のような、値の上限・下限に関する制約は、検査制約では規定しないものとする。

- (2) 表 5 中の [d] ~ [g] に入る適切な数値を答えよ。ここで、空き領域率は 10% とする。

- (3) “顧客基本” テーブルの個人情報保護の強化策について、表 6 のセキュリティ事故 A, B への対策に関する“顧客基本” テーブル実装案の有効性を評価

し,  ,  に “○” 又は “×” を記入せよ。また, セキュリティ事故 C において, “顧客基本” テーブル実装案が有効でないのは, どのような場合か。20字以内で具体的に述べよ。

さらに, 顧客名の暗号化によるセキュリティ強化のトレードオフとして, 性能の悪化, 業務機能の劣化が懸念される。その事象を一つ挙げ, 次の例に倣って 35 字以内で具体的に述べよ。

例：顧客情報の暗号化, 復号のための処理時間が増加する。

設問 2 [データ移行の前提・移行計画・データ移行処理方式設計] について, (1)～(3)に答えよ。

- (1) 表 7 中の  ~  に入れるテーブル名を, 記入済みの例に倣って, 全て答えよ。
- (2) “3. データ移行処理方式設計” 本文中の  ~  に入る適切な数値を答えよ。また, 図 4 中の  ,  に入れる適切な条件を, SQL の述語をそれぞれ一つだけ用いて答えよ。ここで, 現在日付を表す予約語を CURRENT\_DATE とする。
- (3) 表 11 中の  ,  に入る適切な条件を, それぞれ 20 字以内で答えよ。また,  ,  に入る適切なデータ移行方法を答えよ。

設問 3 [データ移行後の検証] について, (1), (2)に答えよ。

- (1) 表 12 中の  ~  に入る適切な字句を答えよ。
- (2) 図 5 の SQL 文では, データが正しく移行されていても, 実行結果の列値がゼロにならない場合がある。その要因を二つ挙げ, それぞれ 20 字以内で述べよ。また, 二つの要因に対する SQL 文の修正内容を, それぞれ 60 字以内で述べよ。