

問6 トレーディングカードの個人間売買サイトの構築に関する次の記述を読んで、設問に答えよ。

S社は、トレーディングカード販売業のチェーンを営む中堅企業である。トレーディングカードを個人から買い取り、販売する事業を営んでいる。トレーディングカードの個人間の売買が盛んな市場環境を受け、個人間の売買を安心かつ手軽に行える取引プラットフォームをサービスとして提供して、安定的な手数料収入を得る新規事業を立ち上げることにした。S社の情報システム部は新規事業の要となる取引プラットフォームのシステム（以下、本システムという）を新規で構築することになり、Tさんがデータベースの設計及び開発を担当することになった。

〔新規事業の業務要件の確認〕

Tさんは、まず、新規事業において実現する業務要件を確認した。新規事業の業務要件（抜粋）を表1に示す。

表1 新規事業の業務要件（抜粋）

項番	業務要件
1	本システムの利用者は個人である。利用者は販売したいトレーディングカードを商品として出品できる。出品した利用者を出品者と呼ぶ。出品する際にはカテゴリ、商品名、商品説明、出品価格、商品状態を登録する。カテゴリはトレーディングカードのブランドやシリーズによって階層化されている。
2	利用者は全ての出品に対してカテゴリ、価格帯（下限価格と上限価格）、商品状態、出品状況、キーワードを指定して検索できる。カテゴリを指定する場合、指定されたカテゴリ及びその下位にある全てのカテゴリの出品が検索の対象となる。キーワードは、商品名及び商品説明を部分一致で検索したい場合に指定する。検索した結果を表示する際に並び順を変更できる。
3	利用者は出品された商品に対して取引を希望する価格（以下、オファー価格という）を出品者に打診できる。この行為をオファーと呼ぶ。オファー価格での取引に出品者が合意した場合、その価格はオファーした利用者と出品者との間での当該商品の出品に対してだけ有効となる。
4	利用者が商品を購入した場合、購入した利用者を購入者と呼ぶ。出品者は配送方法に基づいて送料を算出し、購入者に送料を通知する。購入者は出品価格（オファーが合意済みの状態の場合はオファー価格）と送料の合計金額を支払期日までにS社に対して支払う。一つの取引に関する合計金額を分割して支払うことや、複数の取引に関する合計金額をまとめて支払うことはできない。
5	出品者は購入者の支払が完了したことをS社からの通知で確認した上で、購入者の住所に商品を発送する。一つの取引に関する商品を分割して発送することや、複数の取引に関する商品をまとめて発送することはできない。S社は発送した商品の追跡番号を管理し、配送業者のWebサイトと連携することで、出品者や購入者が商品の配送状況を確認できるようにする。
6	購入者は商品を受領後、商品に問題がないことを確認した上で、受取連絡と出品者の評価を行う。購入者からの評価を受けて、出品者も購入者の評価を行う。
7	購入者と出品者の双方の評価が完了した後に、S社は購入者から入金された代金から手数料を差し引いた金額を出品者に支払う。

表 1 新規事業の業務要件（抜粋）（続き）

項番	業務要件
8	ユーザーはお気に入りのユーザーをフォローできる。フォロー先のユーザーが新たな商品を出品した場合、フォロー元のユーザーは通知を受け取ることができる。

〔概念データモデルの設計〕

Tさんは、表1の業務要件に基づいて、E-R図を用いて本システムの概念データモデルを設計した。本システムの概念データモデル（抜粋）を図1に示す。なお、カテゴリの階層構造は、自己参照の関連を用いて表現する。

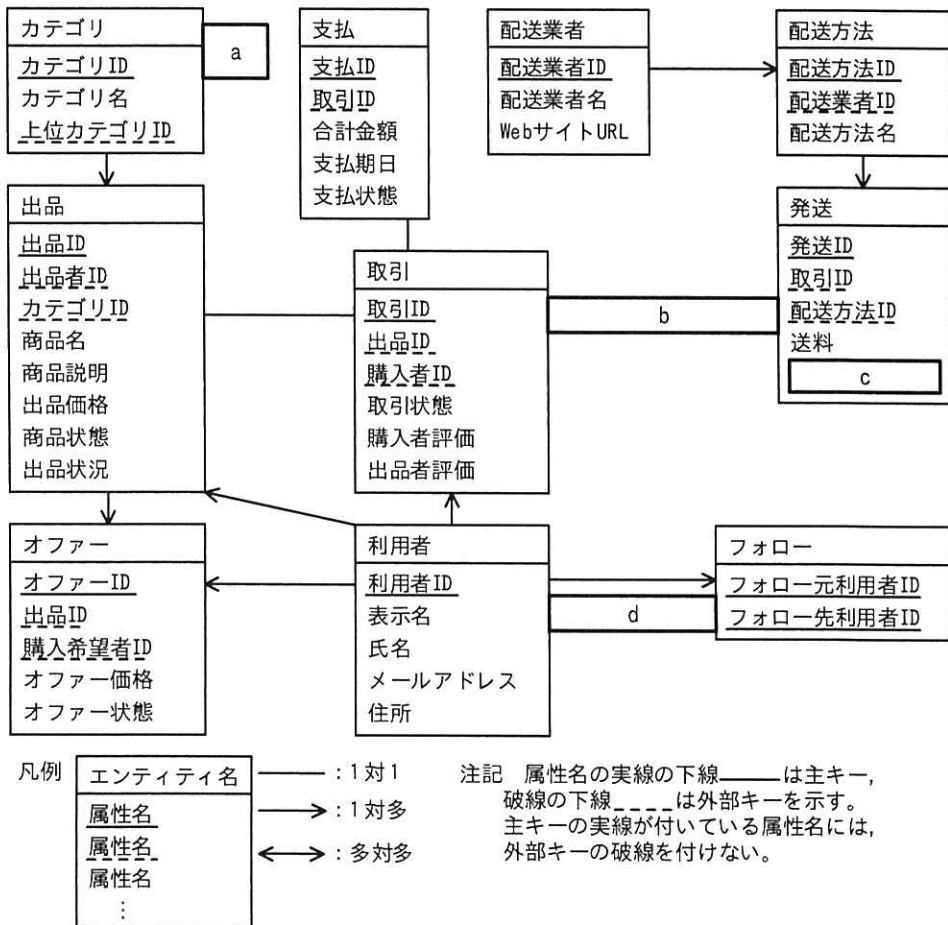


図 1 本システムの概念データモデル（抜粋）

本システムのデータベースでは、E-R図のエンティティ名を表名にし、属性名を列名にして、適切なデータ型で表定義した関係データベースによって、データを管理する。

## [SQL の作成]

Tさんは、表1の項番2の業務要件を実現するための検索のSQL文を作成した。作成したSQL文を図2に示す。なお、“:カテゴリID”、“:下限価格”、“:上限価格”、“:商品状態”、“:出品状況”、“:キーワード”は、該当の値を格納する埋込み変数である。また、最上位であるカテゴリの上位カテゴリIDにはNULLが設定されている。

```

    e 指定カテゴリ(カテゴリID, カテゴリ名, 上位カテゴリID) AS (
    SELECT A.カテゴリID, A.カテゴリ名, A.上位カテゴリID
    FROM カテゴリ A WHERE A.カテゴリID = :カテゴリID
    f
    SELECT B.カテゴリID, B.カテゴリ名, B.上位カテゴリID
    FROM カテゴリ B, 指定カテゴリ C WHERE B.上位カテゴリID = C.カテゴリID
    )
SELECT * FROM 出品
INNER JOIN 指定カテゴリ
ON g
WHERE 出品.出品価格 BETWEEN :下限価格 AND :上限価格
AND 出品.商品状態 = :商品状態
AND 出品.出品状況 = :出品状況
AND (出品.商品名 h OR 出品.商品説明 h )

```

図2 作成したSQL文

## [性能の検証と改善]

Tさんがテストデータを用いて図2のSQL文の実行性能を検証したところ、実行を開始してから検索結果が得られるまでの処理時間が長く、実用的ではないことが判明した。

本システムでは出品される商品の数が膨大であり、利用者が図2のSQL文を頻繁に実行することが予想される。そこで、Tさんはキーワードでの検索が必要な商品名及び商品説明の列には全文検索エンジンを用いるとともに、その他の列に対しては適切なインデックスを設定し、性能上の懸念を解消することを検討した。

インデックスの方式には、B-treeインデックスを採用することにした。Tさんは、各表の表定義を確認し、インデックスを設定すべき列を検討した。出品表の表定義を表2に、カテゴリ表の表定義を表3に示す。

表2及び表3のデータ型欄は、データ型、長さ、精度、位取りを示す。PK欄は主キ

一制約、UK 欄は UNIQUE 制約、非 NULL 欄は非 NULL 制約の指定をするかどうかを示す。指定する場合には Y を、指定しない場合には N が記入されている。ここで、主キーに対しては UNIQUE 制約は指定せず、非 NULL 制約は指定するものとする。カーディナリティ欄は列に多数の異なる値をもつ場合には高を、少数の異なる値をもつ場合には低を記入する。そして、高と低の中間の数の異なる値をもつ場合には中を記入する。データ分布欄は列に含まれる値の確率分布の仮定を示す。

T さんは、①B-tree インデックスの特性を踏まえて、特定の値を指定したときに行数を表全体の 5%以下に絞り込める列だけにインデックスを設定することにした。

表 2 出品表の表定義

項番	列名	データ型	PK	UK	非 NULL	カーディナリティ	データ分布	定義内容
1	出品 ID	INT	Y	N	Y	高	一様分布	1~2000000000 の整数
2	出品者 ID	INT	N	N	Y	高	一様分布	1~2000000000 の整数
3	カテゴリ ID	INT	N	N	Y	中	一様分布	外部キー (カテゴリ)
4	商品名	VARCHAR(255)	N	N	Y	高	一様分布	最大 255 字の文字列
5	商品説明	CLOB	N	N	N	高	一様分布	(省略)
6	出品価格	DECIMAL(10,2)	N	N	Y	高	対数正規分布 (確率密度の最大値: 0.00003)	0.00~99999999.99 の固定小数点数
7	商品状態	SMALLINT	N	N	Y	低	一様分布	1~3 の整数 (1 は未使用, 2 は美品, 3 は傷ありを示す)
8	出品状況	SMALLINT	N	N	Y	低	一様分布	1~3 の整数 (1 は下書き, 2 は出品中, 3 は購入済みを示す)

表 3 カテゴリ表の表定義

項番	列名	データ型	PK	UK	非 NULL	カーディナリティ	データ分布	定義内容
1	カテゴリ ID	INT	Y	N	Y	高	一様分布	1~10000 の整数
2	カテゴリ名	VARCHAR(100)	N	N	Y	高	一様分布	最大 100 字の文字列
3	上位カテゴリ ID	INT	N	N	N	中	一様分布	外部キー (カテゴリ)

T さんは、必要なインデックスを設定後にテストデータを用いて図 2 の SQL 文の実行性能を検証し、実用的な性能であることを確認した。ただし、表 2 及び表 3 のデー

タ分布は新規事業立上げ前の時点における仮定でしかない。今後実際に運用する際にはデータ分布が仮定とは異なる場合があるので、定期的にインデックスを見直すことを申し送り事項の一つとして、本システムのデータベースの設計及び開発を完了した。

設問1 図1中の  ～  に入れる適切なエンティティ間の関連及び属性名を答え、概念データモデルを完成させよ。なお、エンティティ間の関連及び属性名の表記は、図1の凡例に倣うこと。

設問2 図2中の  ～  に入れる適切な字句又は式を答えよ。

設問3 〔性能の検証と改善〕について答えよ。

- (1) 本文中の下線①について、B-tree インデックスの特性として適切なものを解答群の中から三つ選び、記号で答えよ。

解答群

ア インデックスを設定した各列に対する条件を AND 演算子で組み合わせた検索は高速化できるが、NOT 演算子を用いた条件による検索は高速化できない。

イ インデックスを設定した列に対して演算や型変換を行った上で検索条件に指定した場合、検索を高速化できる。

ウ インデックスを設定した列に含まれる値の分布に偏りが無い場合、検索性能が安定する。

エ カーディナリティが低い列にインデックスを設定すると、検索を高速化できる。

オ 行数  $n$  の表において、特定の行を検索するときの計算量は  $O(\log n)$  である。

カ 行数  $n$  の表において、特定の行を挿入するときの計算量は  $O(n)$  である。

キ 等号演算子を用いた条件による検索は高速化できるが、値の範囲を指定した条件による検索は高速化できない。

- (2) 出品表及びカテゴリ表のそれぞれについて、表2及び表3を基に、B-tree インデックスを設定することで図2のSQL文の実行性能の高速化に寄与する列名を全て答えよ。なお、本システムで使用する関係データベースでは、主キーに対して自動的にインデックスが設定される。