

令和1年度秋期 応用情報技術者試験（午後問6）「健康応援システム」 初心者向け復習レポート

1. はじめに：システム概要と出題趣旨の解説

本レポートは、令和1年度秋期応用情報技術者試験の午後問6「健康応援システムの構築」を題材に、データベース設計・運用における重要概念を網羅的に解説する復習ドキュメントです。

事業背景と構築の狙い

ソフトウェアパッケージの開発を行うW社では、デスクワークが多い環境に起因して、従業員が生活習慣病に陥る比率が高く問題となっていました。そこで人事部は、従業員の健康増進を目的に、通信機能をもつ体重計と、歩数や睡眠時間を記録するリストバンド型活動量計（以下、リストバンド）を全社に配布し、これらのデータを統合活用する「健康応援システム」の構築を決定しました。

本問の出題趣旨

本問の出題趣旨は、近年のIoTデバイス普及に伴い生成される膨大なデータをビジネスに活用する背景のもと、**「E-R図やSQL文に関する基本的な理解」 および 「実データからデータモデルの問題箇所を見つけ出し、修正・最適化する能力」**を評価することにあります。

システム構成と主要機能

本システムは、全体として「健康応援データサービス」と「健康応援スマホアプリ」の2つのサブシステムによって構成されています。

- **健康応援データサービス**：本システムのデータを管理するバックエンドプログラム。データの登録・更新・削除用インタフェースおよび定期的なデータ集計機能を担当します。
- **健康応援スマホアプリ**：スマートフォン用のフロントエンドアプリ。体重計やリストバンドと通信を行い、データを健康応援データサービスへ送受信・連携させます。システムが提供する5つの主要機能と、それぞれのデータ処理における意味合いは以下の通りです。

機能名	概要	データベース設計上の戦略的意味合い
手動データ登録機能	電子メールアドレスや身長をスマホ画面から手動登録する。	従業員属性データの維持管理および個人のマスタ構築
データ連携機能	体重計やリストバンドから自動取得したバイタルデータを登録する。	時系列で発生する大量のログデータ（体重・歩数・睡眠）の蓄積
データ公開機能	身長や体重などの情報について、他従業員への閲覧許可（公開情報）を設定する。	従業員ごとのプライバシー制御と共有設定（1対多関係）の保持

月次レポート作成機能	毎月、従業員ごとのBMI、肥満度判定、月間総歩数、平均睡眠時間を集計する。	複数表にまたがる大規模データの定期集計およびレポーティング
歩数対抗戦イベント	部署ごとの従業員一人当たり平均の月間総歩数を競い、利用を促進する。	組織（部署）単位のグループ化集計およびモチベーション向上施策の実現

このように、単に健康データを記録するだけでなく、部署間の対抗イベントや個人の月次集計など多様な利用目的に応えるためには、データの整合性と拡張性を兼ね備えた関係データベースの構築が不可欠となります。次章では、このシステム構築の基盤となる「E-R図の解説」を通じて、エンティティ間の関係性と主キー・外部キーの導出プロセスを詳しく確認していきます。

2. 【設問1】E-R図の読解とリレーションシップの導出

データベース構築において、E-R（Entity-Relationship）図を正確に読み解くことは、システムが保持すべきデータ構造とエンティティ相互の依存関係（リレーションシップ）を正しく定義するために不可欠です。

空欄 a ～ d の導出ロジック解説

E-R図（図1）の空欄 a ～ d を埋める思考プロセスは以下の通りです。

1. 空欄 a（部署と従業員のリレーションシップ）

- **考え方**：業務仕様における「部署ごとの従業員一人当たり平均の月間総歩数を競う」という記述から、1つの部署には複数の従業員が所属し、各従業員は1つの部署に所属することが判明します。
- **導出**：「部署」と「従業員」は **1対多** の関係となります。凡例に従い、1対多の方向を示す記号 → が正解となります。

2. 空欄 b（従業員表の属性・外部キー）

- **考え方**：部署と従業員が1対多の関係にあるため、従業員がどの部署に所属しているかを親表（部署表）から参照して特定する必要があります。
- **導出**：親である部署表の主キーである「部署番号」を、従業員表の外部キーとして保持します。図1の注記により外部キーは破線下線で表すため、破線が引かれた枠に入る属性名は **「部署番号」** となります。

3. 空欄 c（月次レポート表の主キー属性と表記規約）

- **考え方**：月次レポート作成機能では、「毎月、従業員ごとのBMI...を集計する」と記載されています。レポートを一意に識別するためには「誰の」「いつの」レポートであるかを確定させなければなりません。すでに属性として「レポート年月」が定義されているため、誰のデータかを特定する識別子が必要です。
- **導出**：したがって主キーは「従業員番号」と「レポート年月」の複合キーとなります。空欄に入る属性名は **「従業員番号」** です。

- **表記規約の補足**：従業員番号 は親エンティティ（従業員）を参照する外部キーでもあります。図1の注記には「主キーの実線が付いている属性名には、外部キーの破線を付けない」と明記されています。実務や試験表記において「主キー」と「外部キー」の両方の役割を果たす属性が存在する場合、**主キーの表記ルール（実線優先）が適用される**ため、破線ではなく実線下線の枠内に配置されます。

4. 空欄 d（従業員と公開情報のリレーションシップ）

- **考え方**：データ公開機能の仕様において、「身長や体重などのそれぞれの情報について、自分以外の従業員にも閲覧を許可する場合、公開情報として設定する」とあります。一人の従業員は複数の公開設定（身長、体重など）を持つことができ、ある公開設定は一人の従業員に紐付きます。
- **導出**：「従業員」と「公開（中間エンティティ/設定）」は **1対多** の関係になります。従業員から公開へ向かう下向きの矢印 ↓ が正解となります。

設問1 解答一覧

空欄, 解答, 該当概念, 設定の根拠

- a, →, リレーションシップ（1対多）, 1つの部署に複数の従業員が所属するため
 - b, 部署番号, 外部キー（破線下線）, 部署表を参照し所属部署を特定するため
 - c, 従業員番号, 主キーの一部（実線下線）, レポート年月とともに月次データが一意特定されるため（実線優先ルール適用）
 - d, ↓, リレーションシップ（1対多）, 1人の従業員が複数の公開情報設定を行うため
- E-R図によって論理データ構造が可視化された後、システム実装や運用テストの段階で実際のデータ（ログ）を入力した際、データモデルの妥当性が試されることになります。次は、テスト運用で顕在化したトラブルとデータモデルの修正手法について考察します。

3. 【設問3から見るトラブル】実データの問題点とデータモデルの修正

システム開発において、設計段階で想定したモデルと、運用テスト時に投入される「実データ」との間にギャップが生じることは少なくありません。主キー制約（一意性制約）の違反は、データ連携時のエラーを引き起こす代表的な要因です。

CSV睡眠データの分析とエラー原因の特定

リストバンドから取得したCSV形式の睡眠データ（図4）を用いたテストにおいて、データ登録処理時にエラーが発生しました。

"従業員番号", "測定日", "睡眠開始日時", "睡眠終了日時"

```
EMP00001, 2019-10-02, 2019-10-02 22:38:00, 2019-10-03 06:30:00
EMP00001, 2019-10-03, 2019-10-03 23:15:00, 2019-10-04 03:45:00
EMP00001, 2019-10-04, 2019-10-04 04:30:00, 2019-10-04 07:00:00
EMP00001, 2019-10-04, 2019-10-04 23:45:00, 2019-10-05 06:45:00
EMP00001, 2019-10-05, 2019-10-05 23:30:00,
EMP00001, 2019-10-06, 2019-10-06 22:38:00, 2019-10-07 05:45:00
```

提示されたCSVデータの**3行目（2019-10-04 04:30:00開始） と 4行目（2019-10-04 23:45:00開始）**に注目してください。同一の従業員（EMP00001）において、ともに測定日が「2019-10-04」となっています。これは、二度寝や夜勤・仮眠などにより、1日に2回以上睡眠を取得するパターンが発生したことを意味します。

主キー制約違反のメカニズム

従来の「睡眠」表のデータモデル（図1）では、主キーが「従業員番号，測定日」の組み合わせに設定されていました。データベースの主キーには、格納されるレコードが一意（重複しない）でなければならないという「一意性制約（主キー制約）」が存在します。1日に2回以上の睡眠データが存在すると、「EMP00001 + 2019-10-04」というキーを持つレコードが2件挿入されることになり、2件目の挿入（INSERT）時に主キー制約違反が発生して登録エラーとなってしまったのです。

技術的判断：なぜ「睡眠開始日時」を主キーにするのか？

このエラーを解消するためには、主キーの組み合わせを変更して1日複数回の睡眠であっても各レコードを一意に識別できるようにする必要があります。ここで識別子の候補となるのは「睡眠開始日時」または「睡眠終了日時」です。しかし、CSVデータの5行目を見ると、就寝中にリストバンドが外れてしまったことで「睡眠終了日時」が取得できず、値が**NULL（未設定）になっています。主キーの構成要素には「非NULL（NOT NULL）」かつ「確実に一意特定できること」が絶対条件として求められます。したがって、NULLになるリスクがある「睡眠終了日時」ではなく、睡眠をとる限り確実に記録が開始される「睡眠開始日時」**を主キーとして選択するのが技術的に正しい判断となります（「睡眠開始日時」を主キーに含めることで、同じ測定日であっても時刻が異なるため一意性が完全に担保されます）。💡 **コンサルタントの視点：IoTログデータにおける主キー選定リアルタイムで発生するセンサーデータやバイタルログデータを扱う際、「業務上の日付（測定日）」だけで主キーを構成しようとすると、今回のような「同日内の複数発生」や「日付を跨ぐデータ処理」で必ず破綻します。ログデータ設計においては、物理的なタイムスタンプ（開始日時や生成日時）を複合キーに組み込み、かつ機器未着脱や通信エラーによるNULL発生リスクを排除した属性を主キーに選定するのがアンチパターン回避の鉄則です。**

設問3 解答まとめ（記述式対策・文字数検証）

午後試験の記述問題では、指定された文字数制限（30字以内）を厳格に守りつつ、要点を過不足なく記述する能力が求められます。

- 設問3(1) エラーを引き起こしたパターン（下線①）
- 解答例： 1日に2回以上睡眠を取得するパターン （18文字／制限30字以内）
- (別解)： 同一従業員で同じ測定日のデータが複数存在するパターン （26文字／制限30字以内）
- 設問3(2) データモデルの変更内容（下線②）
- 対象の表名： 睡眠
- 変更内容： 主キーを“従業員番号，睡眠開始日時”に変更する。 （26文字／制限30字以内）データモデルの不具合を解消し、すべての睡眠ログを正常に受け入れ

られる構造に正したうえで、次にこのモデル上のデータを集計してレポートを出力する「月次レポート作成機能のSQL実装」を解析していきます。

4. 【設問2】月次レポート作成機能の実装（SQL文の解説）

データベースから必要なデータを抽出し、ビジネスで利用可能な形式に集計・加工する手段がSQLです。ここでは、月次レポートを作成するための実際のSQL文（INSERT/SELECT構文、UPDATE文、集計関数、結合条件）の構造と役割を紐解きます。

処理手順(1)・空欄 e の解説

月次レポート作成の最初のステップは、集計のベースとなる枠組み（レコード）を作成することです。

処理手順(1)

「月次レポート表に従業員番号と集計する対象年月だけがセットされたレコードを挿入する。」これを実現するSQL（図2）は以下の構造をとります。

```
INSERT INTO 月次レポート (従業員番号, レポート年月)
SELECT 従業員番号, :レポート年月
FROM 従業員
```

- **構文の仕組み**：INSERT INTO 表名（列名リスト）SELECT ... FROM ... は、SELECT文の実行結果をそのまま別の表に一括挿入する構文です。
- **空欄 e のロジック**：従業員表（FROM 従業員）から全従業員の「従業員番号」を取得し、プログラムから渡される埋込み変数「:レポート年月」と組み合わせて出力します。したがって、空欄 e には **SELECT 従業員番号, :レポート年月** が入ります。

処理手順(2)④・空欄 f および g の解説

次に、作成された月次レポートレコードに対して、対象年月の月間総歩数を計算して更新（UPDATE）する処理を行います。処理手順(2)④のSQL（図3）は以下の通りです。

```
UPDATE 月次レポート
SET 月間総歩数 =
  (SELECT COALESCE( SUM(歩数.歩数) , 0)
   FROM 歩数
   WHERE 歩数.従業員番号 = 月次レポート.従業員番号
        AND TOYM(歩数.測定日) = :レポート年月 )
WHERE レポート年月 = :レポート年月
```

※注記：文中の TOYM は、問題文の注記にある通り「年月日を年月に変換する関数」です。これにより 歩数.測定日（YYYY-MM-DD）を年月（YYYY-MM）に変換し、埋込み変数 :レポート年月 と一致するレコードを絞り込んでいます。

1. 空欄 f (集計関数)

- **ロジック** : 歩数表から該当月・該当従業員の合計歩数を算出する必要があります。指定された表名と列名を用いて、歩数表 (歩数) の歩数列 (歩数) を合計する集計関数 **SUM(歩数. 歩数)** を適用します。

2. 空欄 g (表の結合条件)

- **ロジック** : サブクエリ (副問合せ) 内で、歩数表のデータと更新対象である月次レポート表のデータが同一従業員のものであることを紐付ける必要があります。そのため、相関サブクエリとして両表の共通キーである従業員番号を等号で結び、**歩数. 従業員番号 = 月次レポート. 従業員番号** (または **月次レポート. 従業員番号 = 歩数. 従業員番号**) と設定します。

3. COALESCE関数の効果と信頼性評価

SQL内の COALESCE(SUM(歩数. 歩数), 0) は、第一引数がNULLの場合に第二引数 (0) を返す関数です。もし、対象月に歩数データが1件も存在しない従業員がいた場合、SUM 関数は NULL を返します。これをそのまま放置するとレポートの値が不定 (NULL) になり、システムの信頼性を損ないます。COALESCE 関数を用いて「歩数データがない場合は 0 歩」として処理することで、欠損値によるトラブルを防ぎ、データ精度の担保に貢献しています。

設問2 解答まとめ

- **e** : **SELECT 従業員番号, :レポート年月**
- **ポイント** : 従業員表から番号を抽出し、埋込み変数を添えてINSERTの入力データを作成する。
- **f** : **SUM(歩数. 歩数)**
- **ポイント** : 歩数表の歩数列を対象として合計値を求める集計関数を正しく記述する。
- **g** : **歩数. 従業員番号 = 月次レポート. 従業員番号**
- **ポイント** : 相関サブクエリにおいて、外側の月次レポート表と内側の歩数表を従業員番号で結合する。集計処理のSQL構文と例外処理 (NULL対策) の解説を終えたところで、最後に本問全体を通じて得られるデータベース設計の教訓を整理しましょう。

5. まとめと応用情報技術者試験突破に向けた学び

本問題 (令和1年度秋期間6) を通じて学んだデータベース設計・運用の本質的なポイントを総括します。

初心者が押さえるべき復習の4大ポイント

1. **E-R図からの正しいリレーションシップ導出** 業務ルール (1対1、1対多、多対多) を読み取り、親表の主キーを子表の外部キーとして参照設定する基本動作を習得する。また、主キーと外部キーを兼ねる属性では主キー表記 (実線) が優先される規約を押さえる。

2. **データモデルの制約（一意性と非NULL）の厳格な適用** 主キーはすべてのレコードを一意に識別できる必要があり、NULLが含まれる可能性のある列（例：リストバンド離脱で未取得になり得る睡眠終了日時）は主キーに含めてはならない。
3. **実データ分析に基づく柔軟なデータモデル見直し** 「1日複数回のデータ発生」など、設計初期の想定を超えるデータパターン（エラー）に直面した際は、実態に合わせて主キーの構成（「睡眠開始日時」を加えた複合キー化）を適切に見直す。
4. **SQLによる正確なデータ集計と例外処理** INSERT INTO ... SELECT ... による一括生成や、相関サブクエリを用いたUPDATE文、ならびに COALESCE 関数や TOYM 関数などを活用したNULL対策・型変換の手法を理解し、データ品質を確保する。

確かな設計力を身につけるために

資格試験の過去問演習においては、単に「解答例の記号やSQL構文を暗記する」だけでは応用力は身につけません。大切なのは、**「なぜその主キー設計では実データ投入時に不具合が起きるのか」「なぜその結合条件と関数が必要になるのか」**という根拠（So What?）を論理的に説明できるようになることです。「仕様変更や想定外のデータ投入にも耐えうるデータモデル設計」という視点を持って復習を重ねることが、応用情報技術者試験の午後問題を突破し、実務でも活躍できる確固たる技術者へと成長するための大きな一歩となります。