帝塚山学院大学
TEZUKAYAMA GAKUIN UNIVERSITY

データベース概論
データベース管理システム

中野秀男
情報メディア学科/ICTセンター長

今日の話

- 復習
  - データベースは集める、整理する、検索する
  - データモデリング
- 少し半構造
- 質問やコメントの回答
- データベース管理システム
  - 高速に検索するために
  - 二分木
  - ハッシュテーブル
  - B-tree

テーブル、レコード、フィールド(再掲)

- テーブル
  - マスターテーブル：店員台帳、お客台帳、料理台帳
    - 店員台帳の主キー: 店員番号
    - お客台帳は？
  - データテーブル：注文データテーブル
    - 注文台帳の主キー: 注文番号 or 注文番号、日付
- レコード
  - 店員、お客、料理
- フィールド
  - 店員：性別、年齢、姓名、勤務日時、時給
  - お客：性別、おおよその年齢
  - 料理：カテゴリー、料金、出す時間帯

リレーションスキーマ, SQL

- キー
  - 主キー 空値がない null
  - 複数の列からなることがある。
    - 一意性と部分集合がキーでない
  - 外部キー: 他のマスターテーブルの主キー
- SQL
  - SQLでは
    - 表(Table), 行(row)、列(column)
  - テーブルの枠組みを定義するSQL文: create
  - データの操作をするSQL文: insert, delete, select
  - データのアクセス権限を指定するSQL文

8.データモデリング

- 8.1 データモデリング
  - 実世界の中でDB化した範囲からデータ項目を抽出
  - 整理
  - DBの適切なデータ構造を決定
- 8.2 データベース設計
- 8.3 実体関連図(ER図)
  - E: Entity(実体), R: Relationship(関連)
  - (1) 実体、(2) 属性、(3) 主キーと外部キー、(4) 実体間の関連
- 履修モデル
- 眼鏡屋の例

半構造

- SQLの限界
  - あいまいな記述ができない
    - 例: メールアドレスが二つある
    - 例: 携帯電話を二台持っているので電話番号が二つある
  - リンクしとけば簡単に検索できるのに、そんな機能がない
- 半構造
  - 自己記述にしておく: データにラベルをつけておく
  - リンクも記述する
- 詳しくはあとの講義で

質問やコメント(1)

- ER図の再度の説明
- 眼鏡屋のER図
- 主キーは最初に入力
- データモデリングの事例の図は良い
- データモデリングは事例があってわかりやすかった
- 教科書でも勉強していきたい
- 教科書の図が案外わかりやすい
- ゆっくりやってくれるので良い
- データモデリングは、どのような商品にも使えるか
- SQLの可能性はどこまでか
- SQLはExcelの拡張版か

質問やコメント(2)

- この講義ではDBは作るのか
- プログラミングでならったのを似ている
- 季節の売れ筋や新しいものはコンビニの目立つところに
- コンビニが進化するとセルフサービスになるか
- スシロー以外で皿にチップは
- アマゾンで1時間以内に届けるサービスは
- アマゾンのプライム会員は普通の会員とどこがちがう
- Googleの自動運転で免許証は。システムのエラーは
- 10年後は想像できない
- VRや3Dモデリング

質問やコメント(3)

- gmailは覗かれているけど問題は
- gmailを就活で使うのはまずいか
- 削除権の法律は良いが大事な情報の隠蔽に
- Tポイントはなんだかいや
- DBをうまく使うと犯罪防止になるか
- 京都府警のITはすごい
- Pepperを買った目的は。何か講義で使うのか
- Pepperの性能は。歌は歌うか？
- Pepperと一緒に写真を撮りたい

質問やコメント(4)

- 若者が少なく老人が増えるのはどうすればいいか
- 人柱とアップデート
- 人が人に対しても表情分析や読心術を学べば予測ができるか
- マイナンバーで税金の捕捉をするが、そもそもなぜ税金が
- マイナンバーの管理の仕方が気になる
- マイナンバーはどのような時に使うのか
- 保険の管理もマイナンバーか
- GOMプレイヤの代わりは

9.データベース管理システム

- SQLのデータベース
  - Oracle
  - SQLserver(Microsoft)
  - MySQL
  - PostgreSQL
- インデックス方式
  - B-treeインデックス
  - ハッシュインデックス

検索

- データは整理されている
  - インデックス
- 検索アルゴリズム
  - 二分木探索(binary search)
  - ハッシュ探索(hash search)
- 二分木探索
  - データを順に並べておく
  - 真ん中のデータが探しているデータか？
    - OK: 検索終了(結果は真ん中のデータ)
    - 小さければ前半のデータセットでやり直し
    - 大きければ後半のデータセットでやり直し
    - データがない(結果は「データなし」)