転載禁止

GoodfindEngineerACADEMY

参加無料　学年不問　20名限定

この夏、基本からWebプログラミングを極める3日間の集中コース

- WebAPIやデザインフレームワークを使った、オリジナルのWebサービス・プロダクトを創る -

GoodfindEngineer

<3 Days= “Programming Bootcamp” >

無線LANにご接続ください

SSID ： slogan-guest (- 5g)　　PASS ： welcomeslogan

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

木村　圭佑

Keisuke Kimura

スローガン株式会社

テクノロジーデザイングループ：エンジニア

日本大学理工学部卒業後、デジタルハリウッドWebエンジニア学科を修了。

小規模のWeb制作会社にエンジニアとして入社し、入社３年で当時社内で最年少開発事業部長に就任。

その後、フリーランスとして独立し、Webシステムを中心に案件を受注。

会社員時代の知人に声をかけられ、スローガン株式会社の描く未来に共感し、Goodfindに参画。

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

Agenda

本日のアジェンダ

1. データベースとは何か
2. データベースの種類
3. データベースの設計
4. SQL入門編
5. CakePHPのModelを使用する

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

Agenda

本日のアジェンダ

1. データベースとは何か
2. データベースの種類
3. データベースの設計
4. SQL入門編
5. CakePHPのModelを使用する

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データの管理方法

ファイルシステム

データベース・
マネジメント・システム

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

ファイルシステム

アプリケーション単位で管理

→ 他のアプリケーションでの利用が不可

異なるファイル間での情報の
重複や関連の考慮せず

→ データの一貫性の欠如

秘密保持や障害回復などの
機能が不足

→ 情報漏洩や紛失が起こりやすい

DBMS

データベースの中に
論理的な定義情報を保持

→ 多様なアプリケーションで利用が可能

データベース内の情報を一元管理

→ データの一貫性の保持

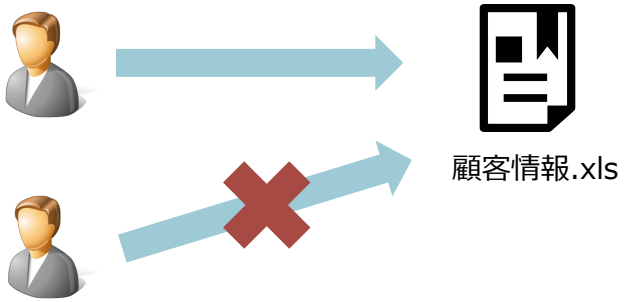
アクセス権限やデータ復旧の
機能が豊富

→ 情報漏洩や紛失が起こりにくい

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

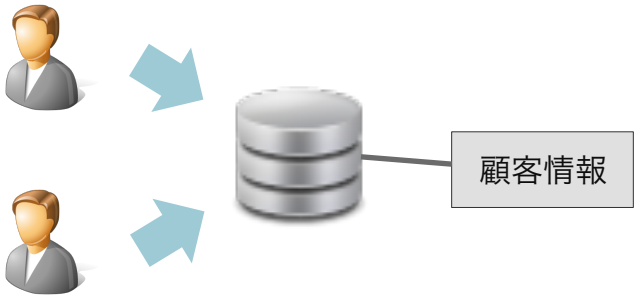
データの共有

ファイルシステム



複数のユーザーが1つのデータを共有して利用する仕組みをもっていないため、一人がアクセスしていると他の人がアクセス出来ない。

DBMS

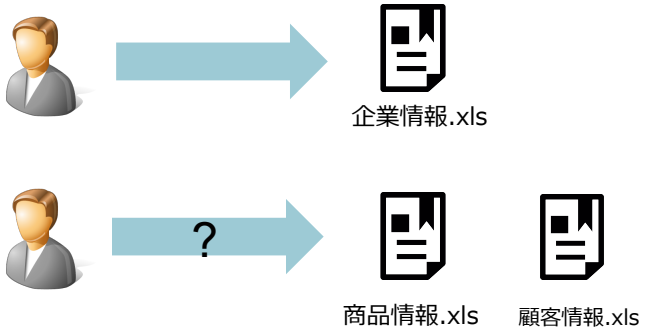


DBMSが複数アクセスを制御してくれるので、複数人による編集が可能。

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

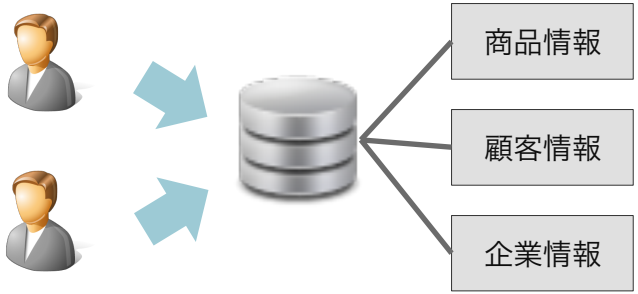
データの位置

ファイルシステム



ファイルの位置がバラバラなため、欲しいデータを探すのが困難。

DBMS



データベースでデータが一元管理されているため、データベースにアクセスすればほしい情報が見つかる。

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データベースとは、
データを**整理・統合**して格納し、
検索しやすく保持するもの。

データの**共有化**やデータ**保護**、データ**検索**、
更新などの処理を簡単に実現することが出来る。

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

Agenda

本日のアジェンダ

1. データベースとは何か
2. データベースの種類
3. データベースの設計
4. SQL入門編
5. SQL応用編

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

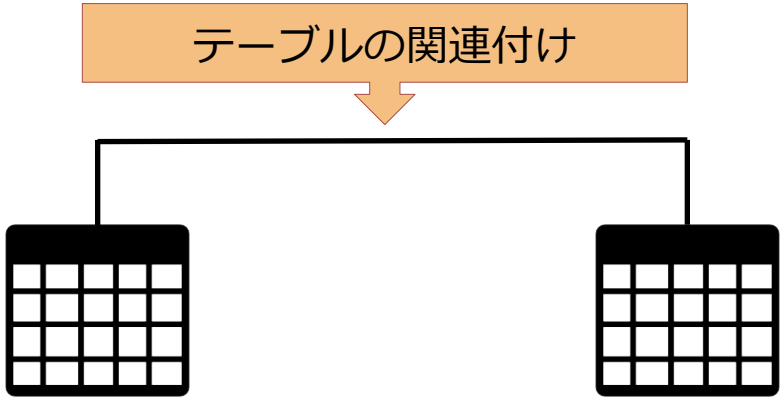
データベースの種類

- **リレーショナル型**データベース
- **ツリー型**データベース
- **ネットワーク型**データベース
- **分散型**データベース
- **オブジェクト指向型**データベース
- **XML**データベース

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

リレーショナル型データベース(RDBMS)

現在最も利用されているデータベース。
テーブルという概念でデータの集合を格納し、各テーブルを**リレーション**によって関連付ける。



Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

リレーショナル型データベースの種類

MySQL

PostgreSQL

Oracle Database

他…

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

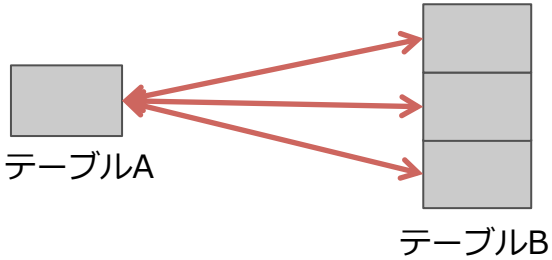
リレーションの種類

1：1の関係

例. ユーザーテーブルとユーザー詳細テーブル

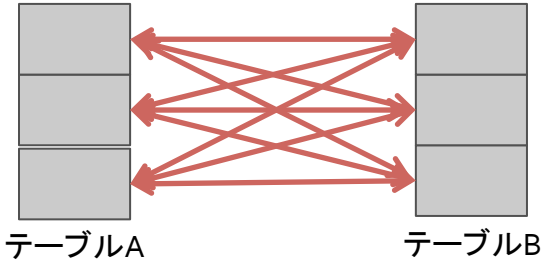
1：多の関係

例. 会社テーブルと顧客テーブル



多：多の関係

例. 顧客テーブルと製品テーブル



Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

Agenda

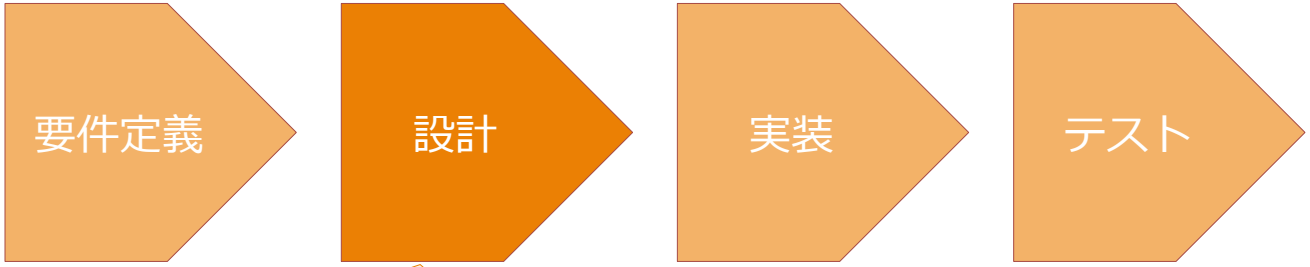
本日のアジェンダ

1. データベースとは何か
2. データベースの種類
3. データベースの設計
4. SQL入門編
5. SQL応用編

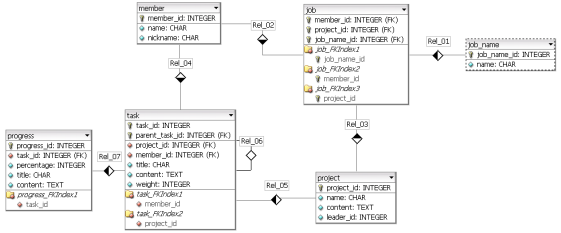
Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

開発工程とデータベース

システム開発の基本工程



要件定義を元にテーブル設計を行う



Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

DBを制するものはWebサービスを制す！

Webサービスの場合はDBの設計を見ただけである程度の**規模感**、**サービスの種類**がわかります。

開発・テストも上流のDB設計に依存する部分が多くあります。

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

DB設計の目的

信頼性の高いデータベースの構築

整合性が維持できる

障害から迅速に**復旧**できる

拡張性の高いデータベースの構築

将来利用する可能性のある**データの蓄積**の実現

ユーザー要求の性能を満たす構築

効率の良いデータアクセス

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データベースの構造

id	username	content	created
1	山田太郎	こんばんは。	2015-06-10 14:49
2	鈴木史朗	さようなら。	2012-06-10 15:49
3	田中裕二	こんにちは。	2012-06-10 16:49
4	中村紀行	Good Bye!	2012-06-10 17:49

レコード

カラム

テーブル

ユーザーの求める**索引**を実現できる

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

カラムと**制約条件**

主キー (PRIMARY KEY)

ユニークキー (UNIQUE)

NOT NULL

外部キー(FOREIGN KEY)

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データ型

数値型　int, double, float …

文字型　varchar, text …

日付型　datetime, timestamp …

バイナリデータ型　binary …

論理型　boolean…

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データベースの**正規化**

データの**重複**をなくし**整合的**にデータを
取り扱えるようにデータベースを設計すること。

第一正規形・第二正規形・第三正規形

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

非正規形

注文番号	顧客番号	顧客名	顧客住所	商品番号１	商品名１	単価１	購入数１	商品番号２	…
P00001	USR001	山田奈美	千葉県小金井市 福山 0-0-00	P0001	どっさりカイロ	300	10	P0002	Windows Versa XP
P00002	USR002	薄井浩	静岡県鎌ヶ谷市 静波 9-8-98	P0001	どっさりカイロ	300	5		
P00003	USR003	日尾晃子	埼玉県新田原市 三次 1-1-11	P0002	Windows Versa XP	21500	2	P0003	サンタのカッコ
P00004	USR001	山田奈美	千葉県小金井市 福山 0-0-00	P0010	豪華おせち	15000	1		

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

第一正規形

注文番号	顧客番号	顧客名	顧客住所
PO0001	USR001	山田奈美	千葉県小金井市福山 0-0-00
PO0002	USR002	薄井浩	静岡県鎌ヶ谷市静波 9-8-98
PO0003	USR003	日尾晃子	埼玉県新田原市三次 1-1-11
PO0004	USR001	山田奈美	千葉県小金井市福山 0-0-00

▲注文テーブル

注文番号	商品番号	商品名	単価	購入数
PO0001	P0001	どっさりカイロ	300	10
PO0001	P0002	Windows Versa XP	21500	1
PO0002	P0001	どっさりカイロ	300	5
PO0003	P0002	Windows Versa XP	21500	2
PO0003	P0003	サンタのカッコ	5000	1
PO0004	P0010	豪華おせち	15000	1

▲注文明細テーブル

参考画像:「正規化ってなに?」（IT pro）

データ内の繰り返し部分をなくす

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

第二正規形

注文番号	顧客番号	顧客名	顧客住所
P00001	USR001	山田奈美	千葉県小金井市福山 0-0-00
P00002	USR002	薄井浩	静岡県鎌ヶ谷市静波 9-8-98
P00003	USR003	日尾晃子	埼玉県新田原市三次 1-1-11
P00004	USR001	山田奈美	千葉県小金井市福山 0-0-00

▲注文テーブル

注文番号	商品番号	購入数
P00001	P0001	10
P00001	P0002	1
P00002	P0001	5
P00003	P0002	2
P00003	P0003	1
P00004	P0010	1

▲注文明細テーブル

商品番号	商品名	単価
P0001	どっさりカイロ	300
P0002	Windows Versa XP	21500
P0003	サンタのカッコ	5000
P0010	豪華おせち	15000

▲商品テーブル

参考画像:「正規化ってなに?」（IT pro）

主キー以外にキー項目に関数従属しているものを除く

＊**関数従属**⇒ ある属性の値により、他の属性の値を一意に識別できること

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

第三正規形

注文番号	顧客番号
P00001	USR001
P00002	USR002
P00003	USR003
P00004	USR001

▲注文テーブル

顧客番号	顧客名	顧客住所
USR001	山田奈美	千葉県小金井市福山 0-0-00
USR002	薄井浩	静岡県鎌ヶ谷市静波 9-8-98
USR003	日尾晃子	埼玉県新田原市三次 1-1-11

▲顧客テーブル

注文番号	商品番号	購入数
P00001	P0001	10
P00001	P0002	1
P00002	P0001	5
P00003	P0002	2
P00003	P0003	1
P00004	P0010	1

▲注文明細テーブル

商品番号	商品名	単価
P0001	どっさりカイロ	300
P0002	Windows Versa XP	21500
P0003	サンタのカッコ	5000
P0010	豪華おせち	15000

▲商品テーブル

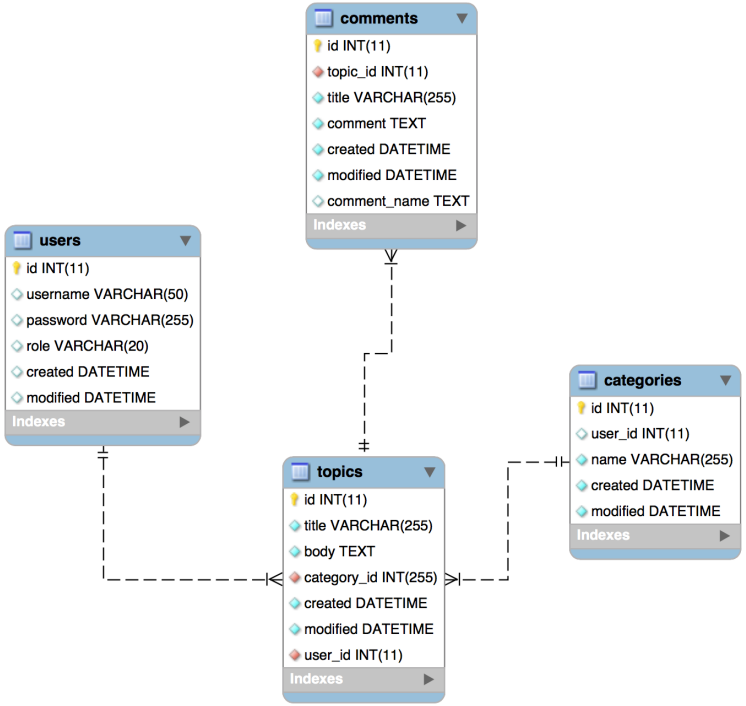
参考画像:「正規化ってなに?」（IT pro）

推移的に関数従属しているものを除く

＊**関数従属**⇒ ある属性の値により、他の属性の値を一意に識別できること

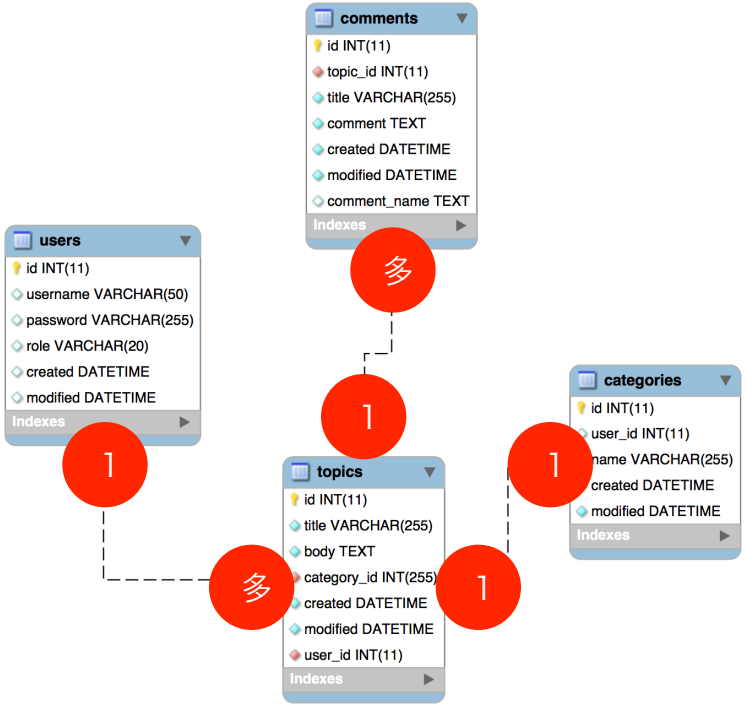
Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

今日のデータベース構造



Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

今日のデータベース構造



Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

Agenda

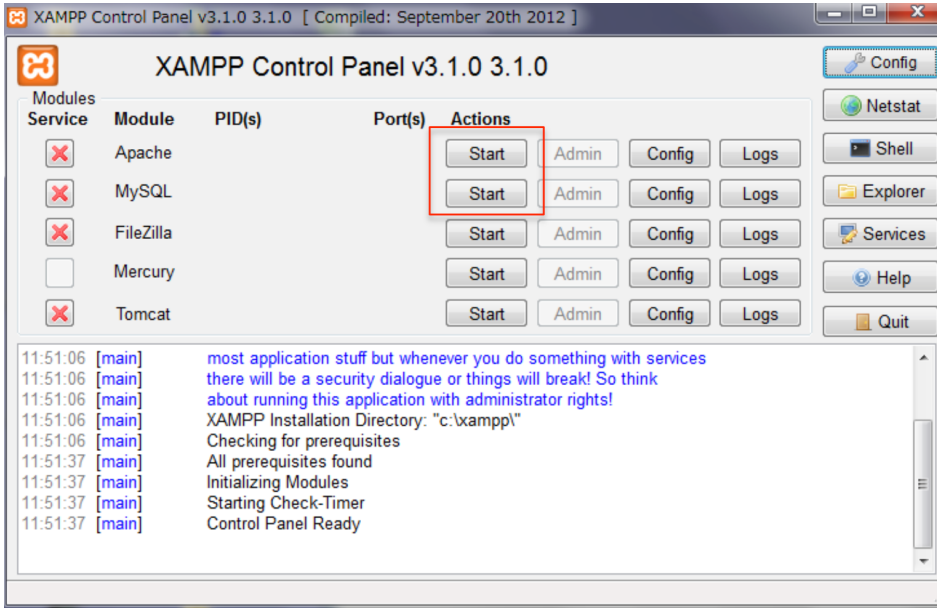
本日のアジェンダ

1. データベースとは何か
2. データベースの種類
3. データベースの設計
4. SQL入門編
5. SQL応用編

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

環境準備

ApacheとMySQLを立ち上げましょう



Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

環境準備

phpmyadminを開きましょう

MACの場合

- スタートページを開く」をクリック
- ブラウザに新しいページがでてきます。
- そのページのphpmyadminタブを選択

Windowsの場合

「http://localhost/phpmyadmin/」にアクセス

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

blog.sqlをインポート

サーバ: localhost » データベース: blog

構造 | SQL | 検索 | クエリ | エクスポート | インポート | 操作 | 特権 | ルーチン | イベント

データベース "blog" へのインポート

インポートするファイル:

ファイルは圧縮されていないもの、もしくは、gzip, bzip2, zip で圧縮されているもの。
圧縮ファイルの名前は**.[フォーマット].[圧縮形式]**で終わっていること。例：**.sql.zip**

アップロードファイル: ファイルを選択 選択されていません (最長: 2,048KiB)

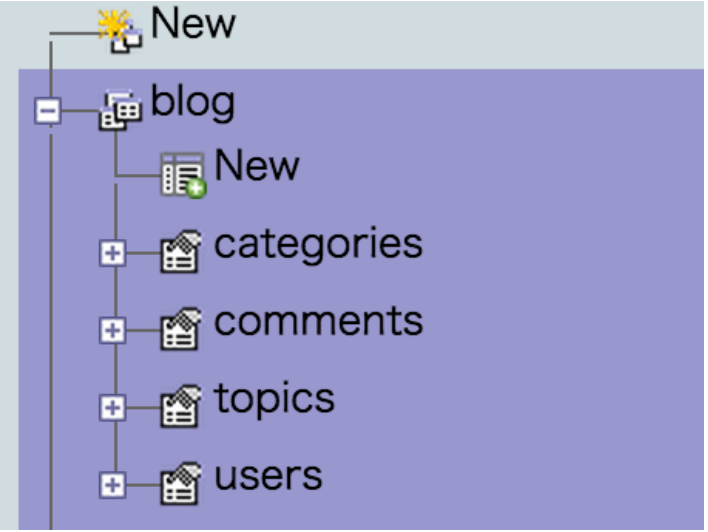
ウェブサーバ上のアップロードディレクトリ **/var/lib/phpMyAdmin/upload/** から選択する： *アップロードファイルがありません*

ファイルの文字セット: utf-8

部分インポート:

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データベースといくつかのテーブルが作成されました。



インポートしたSQLの一部を見てみましょう

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

```
CREATE TABLE `topics` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `title` VARCHAR(255) NOT NULL,
  `body` text COLLATE utf8_unicode_ci NOT NULL,
  `category_id` int(255) NOT NULL,
  `created` datetime NOT NULL,
  `modified` datetime NOT NULL,
  `user_id` int(11) NOT NULL,
  PRIMARY KEY (`id`),
  CONSTRAINT `FK_topics_user_id`
    FOREIGN KEY (`user_id`)
    REFERENCES `blog`.`users` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `FK_posts_category_id`
    FOREIGN KEY (`category_id`)
    REFERENCES `blog`.`categories` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;
```

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データを挿入してみよう

データ挿入にはINSERT文を使う

```
INSERT INTO テーブル名
(
        カラム1,
        カラム2,
        カラム3
) VALUES (
        データ1,
        データ2,
        データ3
);
```

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データを挿入してみよう

categoriesテーブルに「日記」カテゴリーを挿入

```
INSERT INTO categories (
        name,
        created,
        modified
) VALUES (
        '日記',
        now(),
        now()
);
```

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データを挿入してみよう

usersテーブルに以下のユーザーを挿入してみましょう

「山田太郎」

「鈴木一郎」

「佐藤林太郎」

「田中耕太郎」

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データを更新してみよう

データ更新にはUPDATE文を使う

```
UPDATE テーブル名
SET
    カラム名１ = データ１,
    カラム名２ = データ２
WHERE
    条件
```

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データを更新してみよう

usersテーブルのid=1の名前を更新してみよう。

```
UPDATE users
SET
    name = '高橋雄太郎',
    modified = now()
WHERE
    id = 1
```

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データを更新してみよう

usersテーブルの他の人の名前も更新してみましょう。

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データを削除してみよう

データ削除にはDELETE文を使う

```
DELETE FROM
    テーブル名
WHERE
    条件
```

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データを削除してみよう

usersテーブルの id=2の人を削除してみよう

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データを抽出してみよう

データ抽出にはSELECT文を使う

```
SELECT
    カラム名
FROM
    テーブル名
WHERE
    条件
```

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データを抽出してみよう

1. ユーザー一覧を抽出してみよう
2. ユーザーID=1の人のみ抽出してみよう

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

SQL入門まとめ

挿入：INSERT文

更新：UPDATE文

削除：DELETE文

抽出：SELECT文

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

テーブルを連結させる

テーブル連結にはJOINを使う

```
SELECT
    カラム名１
FROM
    テーブル名１
INNER JOIN テーブル名２ ON (
    カラム名 = カラム名
)
```

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

テーブル連結演習 1

以下のように抽出してみよう

title 記事タイトル	body 記事内容	name ユーザー名
ブログタイトル１	ブログ記事１	テスト１太郎
ブログタイトル２	ブログ記事２	テスト１太郎
ブログタイトル３	ブログ記事３	テスト１太郎
ブログタイトル４	ブログ記事４	テスト１太郎
ブログタイトル５	ブログ記事５	テスト１太郎
ブログタイトル６	ブログ記事６	テスト２太郎
ブログタイトル７	ブログ記事７	テスト２太郎
ブログタイトル８	ブログ記事８	テスト３太郎

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

テーブル連結演習 1

```
SELECT
    topics.title, topics.body, users.username
FROM
    topics
INNER JOIN users ON (
    topics.user_id = topics.id
)
```

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

MVCフレームワークにおいてのModelの役割

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

CakePHPの命名規則

基本的にこの命名規則に従って、ファイル名・クラス名・テーブル名を決定する

・ コントローラ名は、**複数形+Controller** [キャメル方式]
例. PostsController ⇄ app/Controller/PostsController.php

・ モデル名は、**単数形** [キャメル方式]
例. Post ⇄ app/Model/Post.php

・ビュー名は、**アンダースコアで区切った小文字**
例. index ⇄ app/View/Posts/index.ctp

・ データベースのテーブル名は、**複数形** (例 posts)
・ サロゲートキー（プライマリーキー）は id
・ 外部キーは テーブル名+id (例 category_id @posts)

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

モデル (Post)

基本的にモデル一つが一つのテーブルに対応する

・ リレーションの定義
- belongsTo, hasOne, hasMany

・ validationの定義
- 「空であってはいけない」「メールアドレス」など、テーブルのカラムごとに制限を加える

・ メソッドの定義
- よく利用するロジックをメソッドとして定義して再利用する
- このメソッドの定義を上手くすると開発効率が飛躍的にUP
- 共通メソッドは AppModelに記述

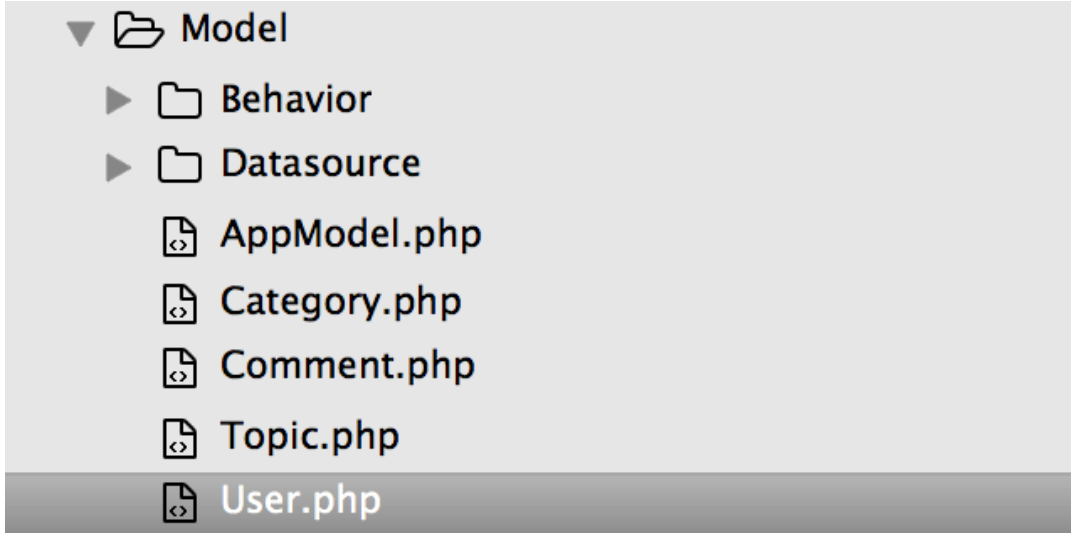
Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

アソシエーション
（モデル同士を繋ぐ）

リレーションシップ	アソシエーション名	例
1 対 1	hasOne	ユーザーは1つのプロフィールを持っている
1 対 多	hasMany	ユーザーは複数のレシピを持っている
多 対 1	belongsTo	レシピはユーザーに属している
多 対 多	hasAndBelongsToMany	レシピは材料を持っており、かつ属している

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

Modelフォルダ内にテーブル名（単数形）.phpを作成



Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

database.php にデータベースへの接続情報を記載する

```
public $default = array(
    'datasource' => 'Database/Mysql',
    'persistent' => false,
    'host' => '127.0.0.1:8889',
    'login' => 'root',
    'password' => 'root',
    'database' => 'blog',
    'prefix' => '',
    'encoding' => 'utf8',
);
```

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データを取得する

Model::find()

を使う

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データを取得する

Model::find()　＝　SELECT

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

カテゴリー一覧を取得する

Category.phpにデータを取得する関数を作る

```
public function getCategories()
  {
     $categories = $this->find('all');
     return $categories;
  }
```

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

カテゴリー一覧を取得する

ControllerからModelの関数を呼び出す（PagesController.php）

```
public function category_list()
{
    $categories = $this->Category->getCategoryList();
}
```

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

カテゴリー一覧を取得する

Viewに値を渡す（PagesController.php）

```
public function category_list()
 {
     $categories = $this->Category->getCategoryList();
     $this->set('caregories', $categories);
 }
```

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

トップページに記事一覧を表示してみよう

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

カテゴリー情報が表示されない

2015-08-05
00:00:00

記事タイトル記事タイトル

テキストテキストテキストテ

Categoriesテーブルと連結

Topics.phpにCategoriesテーブルを連結させる記述を記載

```
public $belongsTo = array(
    'Category' => array(
        'className' => 'Category',
        'foreignKey' => 'category_id',
        'conditions' => '',
        'fields' => '',
        'order' => ''
    )
);
```

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

記事詳細ページを作ってみよう

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

WHERE文はconditiionsで表すことが出来る

```
array(
        'conditions' => array(
            'Topic.id' => $id
        )
);
```

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

データを１レコードのみ取得するばあいは

Model::find(‘first’)

を使う

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

第２引数に、条件（conditionをいれる）

Model::find(‘first’, 条件)

を使う

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.

詳細ページにコメントを出力してみよう

（ヒント１、記事とコメントは１：多の関係）
（ヒント２、１：多のアソシエーションは？）

Copyright © 2006-2015 SLOGAN Inc. All Rights Reserved.