

タイトル	
ここまでの内容を使ってガチャシステムを作ろう！	
レベル	★★★★★★★★

### 【ストーリー】

先輩はどうやら、少し体調がよくなったようです。  
家で作ったプログラムをメールで送ってくれました。

これをもとに、がんばって納期に間に合わせましょう！

### 【先輩からのヒント】

**(1)まずは、ログイン処理から書くといいよ！**

**(2)インスタンスの生成の方法は↓**

**クラス名 インスタンス名 = new コンストラクタの呼び出し:**

**例)Main クラスのインスタンス生成の場合**

**Main main = new Main();**

**(3)メソッドの呼び出しは↓**

**インスタンス名.メソッド名**

**戻り値を受け取る時は、メソッドの前に変数を書く**

**例)Main クラスにある getMessage メソッドを呼び出して、String(文字列)を受け取る場合**

**Main main = new Main();**

**String str = main.getMessage();**

**(4)サーフレットで呼び出すモデルのメソッドがわからないときは、シーケンス図を見てみるといいかも！**

#### (5)DB の接続(コネクションプーリング)は↓のように書くといけるはず

```
//////////////////////////////////////////  
//DBの接続  
String resourceName = "jdbc/MySQL";  
String jndi = "java:comp/env/"+resourceName;  
  
context = new InitialContext();  
  
DataSource dataSource = (DataSource) context.lookup(jndi);  
  
con = dataSource.getConnection();
```

#### (6)List クラスは、そんなに難しくないよ！

List クラスは、指定したタイプのデータを配列みたいにたくさん保持できるよ。

List クラスは List<>の <>の間にリストの型を書くんだよ。

例えば、文字列(String)のリストの場合は List<String>って書くよ。

リストはインスタンスを作って、add メソッドでどんどん追加できるよ！

例)数値型のリストにどんどん値を追加する例

```
List<Integer> list = new ArrayList<Integer>();  
for( int i = 0; i < 10; i++ ){  
    list.add( i );  
}
```

#### (7)DB の接続がうまくできないときは設定ファイルを見てみて！

DB の接続がうまくいかないときは、設定ファイルをみてみよう！

WebContent\META-INF の中にある context.xml が接続の設定ファイルだよ。

特にこの中の、ユーザー名とパスワードが間違っていないかを、まずみてみよう

#### (8)履歴テーブルに INSERT する SQL

多分、こんな感じになるんじゃないかなあ

```
stmt = con.prepareStatement("INSERT INTO rireki_tbl (id,img,reality,insertdate,mail) VALUES(null,?,?,NOW(),?)");  
stmt.setString(1, gatcha.getImg());  
stmt.setInt(2, gatcha.getReality());  
stmt.setString(3, loginBeans.getMail());  
stmt.executeUpdate();
```

ポイントは2つ

1. id は AutoIncrement だから、null を設定してあげるといいよ (MySQL 側で勝手に番号を値を入れてくれる)
2. insertdate には現在時刻がはいるから、MySQL の Now() 関数を使って入れるといいよ！

先輩・・・早く復活してください。

