

Java Web システム ショッピングサイト

システム名	KEN ショッピングサイト
処 理 名	注文情報登録 DAO
プログラム名	OrderDAO
パッケージ	ken.dao

KEN School

処理名	注文情報登録 DAO	クラス名	OrderDAO
業務概要説明			
<p>【処理概要】</p> <ul style="list-style-type: none">● 商品情報、ユーザー情報をデータベースに登録する			

処理名

注文情報登録 DAO

クラス名

OrderDAO

クラス詳細（クラス設計/フィールド設計）

【クラス設計】

項番	項目	値
1	package	ken.dao
2	class/interface/jsp	class
3	アクセス修飾子	public
4	修飾子	—
5	継承元	—
6	実装元	—
7	使用クラス/インタフェース	java.sql.Connection; java.sql.DriverManager; java.sql.PreparedStatement; java.sql.ResultSet; java.sql.SQLException; java.util.ArrayList; ken.bean.Item; ken.bean.User;
8	アノテーション	—

【フィールド設計】

項番	可視性	型	変数名	初期値
1	private	Connection	connection	
2	private	PreperedStatement	p_statement_selectOrderID	
3	private	PreparedStatement	p_statement_selectUserID	
4	private	PreparedStatement	p_statement_insertUser	
5	private	PreparedStatement	p_statement_insertOrder	

処理名	注文情報登録 DAO	クラス名	OrderDAO
-----	------------	------	----------

メソッド・コンストラクタ詳細

【コンストラクタ設計】

コンストラクタ名	OrderDAO
----------	----------

項番	項目	値
1	アクセス修飾子	public
2	修飾子	—
3	戻り値	—
4	引数	—
5	try-catch/throws	ClassNotFoundException, SQLException
6	アノテーション	—

【処理内容】

1. Class クラスの forName メソッドの引数に文字列 "com.mysql.jdbc.Driver" を指定して呼び出す

2. String 型の変数を作成し、"jdbc:mysql://localhost:3306/latte_station?useSSL=false" を代入する

3. String 型の変数を作成し、"root" を代入する

4. String 型の変数を作成し、"root" を代入する

5. DriverManager クラスのクラスメソッドである getConnection() メソッドを呼び出す。その際に、第一引数として 1. で作成した変数を、第二引数として 2. で作成した変数を、第三引数として 3. で作成した変数を設定し、戻り値をフィールド変数 connection に格納する

6. String 型の変数を作成し、"INSERT INTO latte_station.user(user_id,user_name,user_name_kana,post,address,phone,mail) VALUES(?,?,?,?,?,?,?)" を代入する

7. String 型の変数を作成し、"INSERT INTO latte_station.orderItem(order_id,user_id,item_id) VALUES (?,?,?)" を代入する

8. String 型の変数を作成し、"SELECT MAX(order_id) as max_o FROM latte_station.orderitem" を代入する

9. String 型の変数を作成し、"SELECT MAX(user_id) as max_u FROM latte_station.user" を代入する

10. フィールド変数 connection が参照しているオブジェクトの preparedStatement() メソッドを呼び出す。その際に、引数として 6. で作成した変数を設定し、戻り値をフィールド変数 p_statement_insertUser に格納する

11. フィールド変数 connection が参照しているオブジェクトの preparedStatement() メソッドを呼び出す。その際に、引数として 7. で作成した変数を設定し、戻り値をフィールド変数 p_statement_insertOrder に格納する

12. フィールド変数 connection が参照しているオブジェクトの preparedStatement() メソッドを呼び出す。その際に、引数として 8. で作成した変数を設定し、戻り値をフィールド変数 p_statement_selectOrderID に格納する

13. フィールド変数 connection が参照しているオブジェクトの preparedStatement() メソッドを呼び出す。その際に、引数として 9. で作成した変数を設定し、戻り値をフィールド変数 p_statement_selectUserID に代入する

処理名	注文情報登録 DAO	クラス名	OrderDAO
-----	------------	------	----------

メソッド・コンストラクタ詳細

【メソッド設計】

メソッド名	orderRegistration		
-------	-------------------	--	--

項番	項目	値
1	アクセス修飾子	public
2	修飾子	—
3	戻り値	int 型 : オーダーID番号
4	引数	User 型 user : 注文をするユーザー情報
	引数	ArrayList<Item>型 item : 注文する商品情報を保持するリスト
5	try-catch/throws	Exception
6	アノテーション	—

【処理内容】

1. ローカル変数として ResultSet 型の変数 rs_order、rs_user を宣言すると同時に null で初期化する

2. ローカル変数として int 型の変数 orderID を宣言すると同時に 0 で初期化する

3. フィールド変数 p_statement_selectOrderID の executeQuery() メソッドを呼び出し、戻り値をローカル変数 rs_order に格納する

4. もし、ローカル変数 rs_order の next() メソッドの戻り値が true の場合

4.1 ローカル変数 rs_order の getInt() メソッドを呼び出す。その際に、引数に "max_o" を渡し、戻り値をローカル変数 orderID に格納する

4.2 ローカル変数 orderID の値に 1 加算する

5. もし、4. が false の場合

"オーダーに失敗しました" という文字列を引数に渡して Exception クラスのオブジェクトを生成し、throw する

6. フィールド変数 p_statement_selectUserID の executeQuery() メソッドを呼び出し、戻り値をローカル変数 rs_user に格納する

7. ローカル変数として int 型の変数 userID を宣言すると同時に 0 で初期化する

8. もし、ローカル変数 rs_user の next() メソッドの戻り値が true の場合

8.1 ローカル変数 rs_user の getInt() メソッドを呼び出す。その際に、引数として "max_u" を渡し、戻り値をローカル変数 userID に格納する

8.2 ローカル変数 userID の値に 1 加算する

9. もし、8. の結果が false の場合

"オーダーに失敗しました" という文字列を引数に渡して Exception クラスのオブジェクトを生成し、throw する

10. フィールド変数 p_statement_insertUser の setInt() メソッドを呼び出す。その際に、第一引数として 1、第二引数としてローカル変数 userID の値を渡す

11. フィールド変数 p_statement_insertUser の setString() メソッドを呼び出す。その際に、第一引数として 2、第二引数として、このメソッドの引数として受け取ったローカル変数 user の getUsername() メソッドの戻り値を渡す

12. フィールド変数 p_statement_insertUser の setString() メソッドを呼び出す。その際に、第一引数として 3、第二引数として、このメソッドの引数として受け取ったローカル変数 user の getUsername_kana() メソッドの戻り値を渡す

処理名	注文情報登録 DAO	クラス名	OrderDAO
メソッド・コンストラクタ詳細			
【メソッド設計】			
メソッド名	orderRegistration		
【処理内容 つづき】			
<div>13. フィールド変数 p_statement_insertUser の setString() メソッドを呼び出す。その際に、第一引数として 4、第二引数として、このメソッドの引数として受け取ったローカル変数 user の getPostCode() メソッドの戻り値を渡す</div> <div>14. フィールド変数 p_statement_insertUser の setString() メソッドを呼び出す。その際に、第一引数として 5、第二引数としてローカル変数 user の getAddress() メソッドの戻り値を渡す</div> <div>15. フィールド変数 p_statement_insertUser の setString() メソッドを呼び出す。その際に、第一引数として 6、第二引数として、このメソッドの引数として受け取ったローカル変数 user の getPhoneNumber() メソッドの戻り値を渡す。</div> <div>16. フィールド変数 p_statement_insertUser の setString() メソッドを呼び出す。その際に、第一引数として 7、第二引数として、このメソッドの引数として受け取ったローカル変数 user の getMailAddress() メソッドの戻り値を渡す</div> <div>17. フィールド変数 p_statement_insertUser の executeUpdate() メソッドを呼び出す</div> <div>18. 引数で受けとった ArrayList クラスオブジェクトの size() メソッドの戻り値分、以下の処理を繰り返す</div> <div><div>18.1 フィールド変数 p_statement_insertOrder の setInt() メソッドを呼び出す。その際に、第一引数として 1、第二引数としてローカル変数 orderID の値を渡す</div><div>18.2 フィールド変数 p_statement_insertOrder の setInt() メソッドを呼び出す。その際に、第一引数として 2、第二引数としてローカル変数 userID の値を渡す</div><div>18.3 フィールド変数 p_statement_insertOrder の setInt() メソッドを呼び出す。その際に、第一引数として 3、第二引数として引数で受け取っている ArrayList クラスオブジェクトの get() メソッドの引数に、現在のカウンタ変数の値を渡し、戻り値を Item クラス型に変換した値を渡す</div></div> <div>19. 3. ~ 18. を try 句で囲い、25. までの処理を finally 句で囲う</div> <div>20. フィールド変数 rs_order が null でない場合</div> <div><div>フィールド変数 rs_order の close() メソッドを呼び出す</div></div> <div>21. フィールド変数 rs_user が null でない場合</div> <div><div>フィールド変数 rs_user の close() メソッドを呼び出す</div></div> <div>22. フィールド変数 p_statement_insertUser が null でない場合</div> <div><div>フィールド変数 p_statement_insertUser の close() メソッドを呼び出す</div></div> <div>23. フィールド変数 p_statement_insertOrder が null でない場合</div> <div><div>フィールド変数 p_statement_insertOrder の close() メソッドを呼び出す</div></div> <div>24. フィールド変数 p_statement_selectOrderID が null でない場合</div> <div><div>フィールド変数 p_statement_selectOrderID の close() メソッドを呼び出す</div></div> <div>25. フィールド変数 connection が null でない場合</div> <div><div>フィールド変数 connection の close() メソッドを呼び出して、データベースを切断する</div></div> <div>26. ローカル変数 orderID の値を戻り値として返す</div>			