PDO Prepared Statements

Prepared statements are very useful against SQL injections.

A prepared statement is a template for executing one or more SQL queries against the database.

The idea behind prepared statements is that, with queries that use the syntax but different values.

It is much faster to pre-process the syntax once and then execute it several times using different parameters.

Positional Placeholder & Named Placeholder

Positional placeholders are quick to write, they may become a source for hard-to-track errors, especially when you change the query columns. To protect yourself against this, you can use the so- called "**named placeholder**" which consists of a descriptive names preceded by colon (**:**), instead of question marks (**?**).

Positional Placeholder

```php
$sql = $handler->prepare("INSERT INTO `election_login_details`(user_id, password, category) VALUES(?, ?, ?)");

    $sql->bindValue(1, 'Jime');
    $sql->bindValue(2, 'piccolo11');
    $sql->bindValue(3, 2);
        $sql->execute();
```

Named Placeholder

```php
$sql = $handler->prepare("INSERT INTO `election_login_details`(user_id, password, category) VALUES(:user_id, :password, :category)");
    $sql->bindValue(":user_id", 'Abba');
    $sql->bindValue(":password", 'picd999l3o11');
    $sql->bindValue(":category", 2);
        $sql->execute();
            echo 'Records Stored...';
```