Prepared statements in PHP 5 can be used to execute SQL statements that may contain user input or variables. Prepared statements help to avoid SQL injection attacks by escaping special characters in the user input or variables.

Here is an example of using prepared statements in PHP 5:

php

```
// Connect to the database
$conn = mysqli_connect("localhost", "username", "password", "database");

// Prepare a statement
$stmt = mysqli_prepare($conn, "SELECT * FROM users WHERE username = ?");

// Bind parameters to the statement
$username = "john";
mysqli_stmt_bind_param($stmt, "s", $username);

// Execute the statement
mysqli_stmt_execute($stmt);

// Get the results
$result = mysqli_stmt_get_result($stmt);

// Loop through the results
while ($row = mysqli_fetch_assoc($result)) {
    // Do something with each row
}

// Close the statement
mysqli_stmt_close($stmt);

// Close the database connection
mysqli_close($conn);
```

In this example, we first connect to the database using mysqli_connect(). We then prepare a statement using mysqli_prepare() that selects all rows from the users table where the username column matches a parameter. We bind a value to the parameter using mysqli_stmt_bind_param(), and execute the statement using mysqli_stmt_execute(). We get the results using mysqli_stmt_get_result(), and loop through them using mysqli_fetch_assoc(). Finally, we close the statement and the database connection using mysqli_stmt_close() and mysqli_close(), respectively