```
1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel;
4  using System.Data;                          // Do not overwrite this code
5  using System.Data.SqlClient;                // Insert this line as per sample
6  using System.Text.RegularExpressions;       // Insert this line as per sample
7  using System.Drawing;
8  using System.Linq;
9  using System.Text;
10 using System.Threading.Tasks;
11 using System.Windows.Forms;
12
13 namespace SimpleDataApp
14 {
15     public partial class FillOrCancel : Form
16     {
17         // VARIABLES START
18         // Storage for the order ID value.
19         private int parsedOrderID;
20
21         /// <summary>
22         /// Verifies that an order ID is present and contains valid
              characters.
23         /// </summary>
24         private bool IsOrderIDValid()
25         {
26             // Check for input in the Order ID text box.
27             if (txtOrderID.Text == "")
28             {
29                 MessageBox.Show("Please specify the Order ID.");
30                 return false;
31             }
32
33             // Check for characters other than integers.
34             else if (Regex.IsMatch(txtOrderID.Text, @"^\D*$"))
35             {
36                 // Show message and clear input.
37                 MessageBox.Show("Customer ID must contain only numbers.");
38                 txtOrderID.Clear();
39                 return false;
40             }
41             else
42             {
43                 // Convert the text in the text box to an integer to send to
                      the database.
44                 parsedOrderID = Int32.Parse(txtOrderID.Text);
45                 return true;
46             }
47         }
48         // VARIABLES END
49
50         public FillOrCancel()
51         {
52             InitializeComponent();
53         }
54
```

```
55          /// <summary>
56          /// Executes a t-SQL SELECT statement to obtain order data for a
              specified
57          /// order ID, then displays it in the DataGridView on the form.
58          /// </summary>
59          private void btnFindByOrderID_Click(object sender, EventArgs e)
60          {
61              if (IsOrderIDValid())
62              {
63                  using (SqlConnection connection = new SqlConnection
                      (Properties.Settings.Default.connString))
64                  {
65                      // Define a t-SQL query string that has a parameter for
                      orderID.
66                      const string sql = "SELECT * FROM Sales.Orders WHERE
                      orderID = @orderID";
67
68                      // Create a SqlCommand object.
69                      using (SqlCommand sqlCommand = new SqlCommand(sql,
                      connection))
70                      {
71                          // Define the @orderID parameter and set its value.
72                          sqlCommand.Parameters.Add(new SqlParameter("@orderID",
                       SqlDbType.Int));
73                          sqlCommand.Parameters["@orderID"].Value =
                      parsedOrderID;
74
75                          try
76                          {
77                              connection.Open();
78
79                              // Run the query by calling ExecuteReader().
80                              using (SqlDataReader dataReader =
                      sqlCommand.ExecuteReader())
81                              {
82                                  // Create a data table to hold the retrieved
                      data.
83                                  DataTable dataTable = new DataTable();
84
85                                  // Load the data from SqlDataReader into the
                      data table.
86                                  dataTable.Load(dataReader);
87
88                                  // Display the data from the data table in the
                       data grid view.
89                                  this.dgvCustomerOrders.DataSource = dataTable;
90
91                                  // Close the SqlDataReader.
92                                  dataReader.Close();
93                              }
94                          }
95                          catch
96                          {
97                              MessageBox.Show("The requested order could not be
                      loaded into the form.");
98                          }
```

```
 99                          finally
100                          {
101                              // Close the connection.
102                              connection.Close();
103                          }
104                      }
105                  }
106              }
107          }
108
109          /// <summary>
110          /// Cancels an order by calling the Sales.uspCancelOrder
111          /// stored procedure on the database.
112          /// </summary>
113          private void btnCancelOrder_Click(object sender, EventArgs e)
114          {
115              if (IsOrderIDValid())
116              {
117                  // Create the connection.
118                  using (SqlConnection connection = new SqlConnection
                       (Properties.Settings.Default.connString))
119                  {
120                      // Create the SqlCommand object and identify it as a
                     stored procedure.
121                      using (SqlCommand sqlCommand = new SqlCommand
                     ("Sales.uspCancelOrder", connection))
122                      {
123                          sqlCommand.CommandType = CommandType.StoredProcedure;
124
125                          // Add the order ID input parameter for the stored
                     procedure.
126                          sqlCommand.Parameters.Add(new SqlParameter("@orderID",
                      SqlDbType.Int));
127                          sqlCommand.Parameters["@orderID"].Value =
                     parsedOrderID;
128
129                          try
130                          {
131                              // Open the connection.
132                              connection.Open();
133
134                              // Run the command to execute the stored
                     procedure.
135                              sqlCommand.ExecuteNonQuery();
136                          }
137                          catch
138                          {
139                              MessageBox.Show("The cancel operation was not
                     completed.");
140                          }
141                          finally
142                          {
143                              // Close connection.
144                              connection.Close();
145                          }
146                      }
```

```
147                 }
148             }
149         }
150
151         /// <summary>
152         /// Fills an order by calling the Sales.uspFillOrder stored
153         /// procedure on the database.
154         /// </summary>
155         private void btnFillOrder_Click(object sender, EventArgs e)
156         {
157             if (IsOrderIDValid())
158             {
159                 // Create the connection.
160                 using (SqlConnection connection = new SqlConnection
                      (Properties.Settings.Default.connString))
161                 {
162                     // Create command and identify it as a stored procedure.
163                     using (SqlCommand sqlCommand = new SqlCommand
                          ("Sales.uspFillOrder", connection))
164                     {
165                         sqlCommand.CommandType = CommandType.StoredProcedure;
166
167                         // Add the order ID input parameter for the stored
                          procedure.
168                         sqlCommand.Parameters.Add(new SqlParameter("@orderID",
                           SqlDbType.Int));
169                         sqlCommand.Parameters["@orderID"].Value =
                          parsedOrderID;
170
171                         // Add the filled date input parameter for the stored
                          procedure.
172                         sqlCommand.Parameters.Add(new SqlParameter
                          ("@FilledDate", SqlDbType.DateTime, 8));
173                         sqlCommand.Parameters["@FilledDate"].Value =
                          dtpFillDate.Value;
174
175                         try
176                         {
177                             connection.Open();
178
179                             // Execute the stored procedure.
180                             sqlCommand.ExecuteNonQuery();
181                         }
182                         catch
183                         {
184                             MessageBox.Show("The fill operation was not
                              completed.");
185                         }
186                         finally
187                         {
188                             // Close the connection.
189                             connection.Close();
190                         }
191                     }
192                 }
193             }
```

```
194         }
195
196         /// <summary>
197         /// Closes the form.
198         /// </summary>
199         private void btnFinishUpdates_Click(object sender, EventArgs e)
200         {
201             this.Close();
202         }
203     }
204 }
205
```