Connecting and Submitting Queries to a Database



Paul D. Sheriff
BUSINESS SOLUTIONS ARCHITECT, FAIRWAY TECHNOLOGIES, INC.
www.fairwaytech.com psheriff@fairwaytech.com



Module Goals



Create a connection to a database

Scalar queries

Action queries

Output parameters

Transaction processing



Connections



Connection Class

Pass in connection string

Open Connection

```
var cnn = new
SqlConnection(cnnString);
```

```
cnn.Open();
```



Connection Class

Close when done

Dispose of unmanaged resources

cnn.Close();

cnn.Dispose();





Connect to a database

Get connection information





Employ the using block





Catch error information





connectionstrings.com



Commands without Parameters



Command Class

Submit query to database

Call appropriate method

```
sql = "INSERT INTO Product...";
var cmd = new
    SqlCommand(sql, cnn)
```

```
cmd.ExecuteNonQuery();
cmd.ExecuteScalar();
```





Scalar value retrieval





Action command



Commands with Parameters



Parameter Class

Pass argument(s) to SQL statement

Supports OUTPUT parameters

```
cmd.Parameters.Add(new
SqlParameter {
    ParameterName = "@ProductId",
    Value = 1,
    DbType = DbType.Int32,
    Direction =
        ParameterDirection.Output
});
```





Scalar value retrieval





Action command





Action command with output parameter



Transactions



Transactions

More than one statement to submit to database

All or nothing

Commit or Rollback





Transaction processing



Summary



Open connection prior to querying

Can add input and output parameters

Transaction support

Utilize using blocks

Add error handling





Coming up in the next module...

Using a DataReader
Build generic list of entity
objects

Use extension methods

