

Programmazione C#

Lezione 6

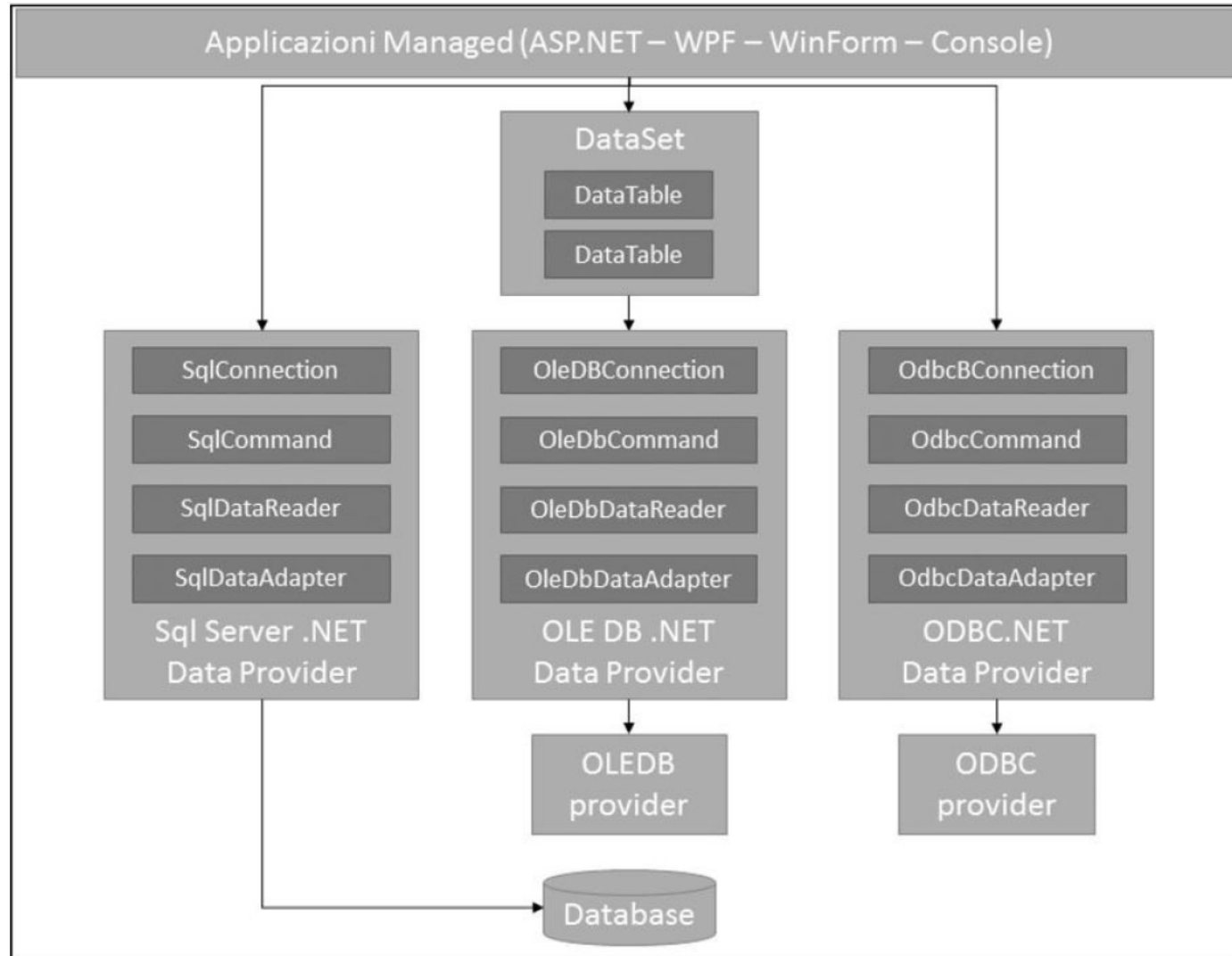
Connessione al DB

Entity Framework



e-lios
e-Linking online Systems

Connessione al DB



Connessione al DB

```
SqlConnection conn = new SqlConnection(connectionString);
SqlCommand SQL = new SqlCommand();
SQL.Connection = conn;
SqlDataAdapter da = new SqlDataAdapter(SQL);

ex_query += @"
    SELECT DESCRIZIONE
    FROM AZ_SETTORE_ATTIVITA
    WHERE DESCRIZIONE LIKE @sec
";

SQL.Parameters.AddWithValue("sec", sec);
SQL.CommandText = ex_query;

DataView dv = new DataView(new DataTable());
da.Fill(dv.Table);
```

```
string connectionString = "...";
using(SqlConnection connection = new SqlConnection(connectionString))
{
    connection.Open();
    // ...
}
```

```
SqlCommand cmd = new SqlCommand("SELECT * FROM Products");
SqlDataReader reader = cmd.ExecuteReader();
try
{
    while(reader.Read())
    {
        // Viene utilizzata la proprietà indexer
        int productID = (int)reader["ProductID"];
        // ProductName è il secondo campo del record
        string productName = reader.GetString(1);
        // ...
    }
}
finally
{
    // Chiusura del data reader
    reader.Close();
}
```

Connessione al DB

```
SqlConnection conn = new SqlConnection(connectionString);
SqlCommand SQL = new SqlCommand();
SQL.Connection = conn;
```

```
ex_query += @"
    INSERT INTO AZ_SETTORE_ATTIVITA
        (DESCRIZIONE)
    VALUES
        (@settore)
";
```

```
SQL.CommandText = ex_query;
SQL.Parameters.AddWithValue("settore", settore);
```

```
conn.Open();
int a = SQL.ExecuteNonQuery();
conn.Close();
```

```
List<SSD> ssd = new List<SSD>();
```

```
OracleConnection con = new OracleConnection();
con.ConnectionString = ConfigurationManager.ConnectionStrings["esse3ConnectionString"].ConnectionString;
con.Open();
```

```
OracleCommand cmd = new OracleCommand();
cmd.Connection = con;
cmd.CommandText = @"
    select ID_SSD, NOME_SETTORE
    from TABELLA
";
```

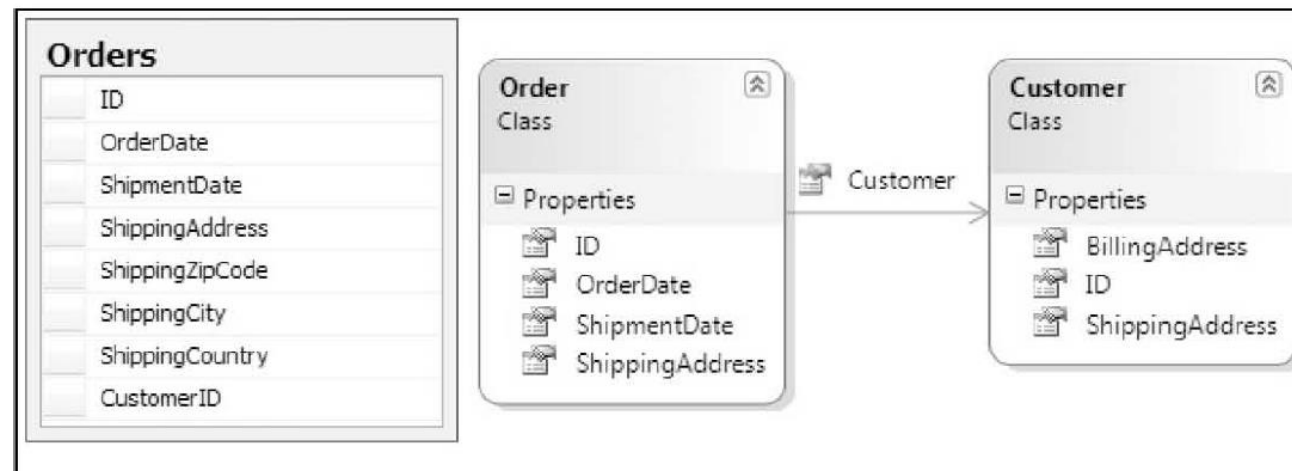
```
cmd.CommandType = System.Data.CommandType.Text;
OracleDataReader dr = cmd.ExecuteReader();
```

```
while (dr.Read())
{
    SSD s = new SSD();
    s.ID_SSD = dr["ID_SSD"].ToString();
    s.NOME_SETTORE = dr["NOME_SETTORE"].ToString();
    ssd.Add(s);
}
```

```
con.Dispose();
```

Entity Framework

1. Un ORM è un «Mapper tra Oggetti e Relazioni»
2. Code-First o Data Base-First
3. L'ORM nativo in .NET è Entity Framework



Entity Framework

```
public enum Shippers : int
{
    SpeedyExpress = 1,
    UnitedPackage = 2,
    FederalShipping = 3
}

public class AddressInfo
{
    public string Address { get; set; }
    public string City { get; set; }
    public string Region { get; set; }
    public string PostalCode { get; set; }
    public string Country { get; set; }
}

public class Customer
{
    public Customer()
    {
        Orders = new HashSet<Order>();
        Address = new AddressInfo();
    }

    public string CustomerID { get; set; }
    public string CompanyName { get; set; }
    public string ContactName { get; set; }
    public string ContactTitle { get; set; }
    public string Phone { get; set; }
    public string Fax { get; set; }
    public AddressInfo Address { get; set; }
    public virtual ICollection<Order> Orders { get; set; }
}
```

```
public class northwindEntities : DbContext
{
    public DbSet<Customer> Customers { get; set; }
    public DbSet<Order_Detail> Order_Details { get; set; }
    public DbSet<Order> Orders { get; set; }
}
```

Entity Framework

Attributo	Applicabile su	Scopo
Table	Classe	Specifica la tabella su cui la classe mappa
Key	Proprietà	Specifica che la proprietà fa parte della chiave primaria
Column	Proprietà	Specifica il nome e il tipo della colonna su cui la proprietà mappa
DatabaseGenerated	Proprietà	Specifica se la colonna è su cui la proprietà mappa è un'identity, calcolata o normale
Required	Proprietà	Specifica che la proprietà è obbligatoria
StringLength	Proprietà	Specifica la lunghezza massima della proprietà

```
modelBuilder.Entity<Order>().Property(p => p.CustomerID).HasMaxLength(5).IsFixedLength();  
modelBuilder.Entity<Order>().Property(p => p.ShipName).HasMaxLength(40);  
modelBuilder.Entity<Order>().Property(p => p.ShipAddress.Address).HasColumnName("ShipAddress").HasMaxLength(60);  
modelBuilder.Entity<Order>().Property(p => p.ShipAddress.City).HasColumnName("ShipCity").HasMaxLength(15);  
modelBuilder.Entity<Order>().Property(p => p.ShipAddress.Country).HasColumnName("ShipCountry").HasMaxLength(15);  
modelBuilder.Entity<Order>().Property(p => p.ShipAddress.PostalCode).HasColumnName("ShipPostalCode").HasMaxLength(10);  
modelBuilder.Entity<Order>().Property(p => p.ShipAddress.Region).HasColumnName("ShipRegion").HasMaxLength(15);
```


Entity Framework: Leggere e Scrivere

Query attraverso LINQ!!!

```
public class northwindEntities : DbContext
{
    public DbSet<Customer> Customers { get; set; }
    public DbSet<Order_Detail> Order_Details { get; set; }
    public DbSet<Order> Orders { get; set; }
}
```

```
using (var ctx = new NorthwindEntities())
{
    var customers1 = from c in ctx.Customers
                    where c.Address.Country == "Italy"
                    select c;

    var customers2 = ctx.Customers.Where(c => c.Address.Country ==
    "Italy");
}
```

Entity Framework: Leggere e Scrivere

```
using (NorthwindEntities ctx = new NorthwindEntities())
{
    var c = new Customer
    {
        CustomerID = "STEM0",
        CompanyName = "Stefano Mostarda",
        ContactName = "Stefano Mostarda",
        Address = new AddressInfo
        {
            Address = "Via Del Corso 14",
            City = "Roma",
            Country = "Italy",
            PostalCode = "00100",
            Region = "Lazio"
        },
        ContactTitle = "Sig",
        Phone = "000000",
        Fax = "000000"
    };
    ctx.Customers.Add(c);
    ctx.SaveChanges();
}
```

- Unchanged: l'oggetto non è stato modificato;
- Modified: qualche proprietà semplice dell'oggetto è stata modificata;
- Added: l'oggetto è stato marcato per l'aggiunta sul database;
- Deleted: l'oggetto è stato marcato per la cancellazione dal database.

Entity Framework: Leggere e Scrivere



Fine Lezione 6

DOMANDE?