# Esercitazione di Fine Settimana – Week 4

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Nome | Michela |
|  |  | Cognome | Murtas |
|  |  | Data | 08/10/2021 |

Leggete attentamente ogni domanda e argomentare quanto più possibile fornendo anche degli esempi.

1. Descrivere le modalità di utilizzo di ADO.NET

ADO.NET ha due diverse modalità di utilizzo:

**Connected Mode** che permette l’accesso di sola lettura ai dati.

**Disconnected Mode** che permette la manipolazione dei dati recuperati per poterli poi riconciliare con l’originale in seguito.

1. Quali sono i metodi di esecuzione della classe DbCommand disponibili e in quali casi vanno utilizzati?

I metodi di esecuzione della classe DBCommand sono:

**ExecuteReader**, attraverso il quale viene restituita la lettura dei dati

**ExecuteNonQuery**, dove viene eseguito un programma

**ExecuteScalar**, dove è possibile far restituire il primo valore della prima riga presente nel DB

1. Descrivere l’utilizzo della classe DataAdapter

Il DataAdapter funge da ponte tra la DisconnectedMode e la ConnectedMode, e da la possibilità di recuperare e utilizzare i dati attraverso il DataSet.

1. Come si definisce il modello dati in Entity Framework?

Entity Framework è il framework ORM (object-relational mapping) messo a disposizione dal .NET Framework e permette di utilizzare i dati contenuti nel database relazionale come ad un insieme di oggetti.

1. Descrivere l’utilizzo delle Migration e i vantaggi che ne derivano

Le Migration consentono l’utilizzo dell’approccio code-first per aggiornare il database e ci consentono di avere un maggiore controllo del codice.

**Esercitazione Pratica**

Realizzare un sistema di gestione delle spese che si basi su:

* Un database **GestioneSpese** (SQL Server), costituito dalle tabelle
  + **Spese**
    - *Id* (int, PK, auto-incrementale)
    - *Data* (datetime)
    - *CategoriaId* (int, FK)
    - *Descrizione* (varchar(500))
    - *Utente* (varchar(100))
    - *Importo* (decimal)
    - *Approvato* (bit)
  + **Categorie**
    - *Id* (int, PK, auto-incrementale)
    - *Categoria* (varchar(100))
* Una **Console app** che consenta di:
  + Inserire nuove Spese
  + Approvare le Spese esistenti (impostare il campo *Approvato*)
  + Cancellare le Spese esistenti
  + Mostri
    - l'elenco delle Spese Approvate
    - L'elenco delle Spese di uno specifico Utente
    - Il totale delle Spese per Categoria

VINCOLI TECNICI

* Utilizzare Entity Framework
* Utilizzare l'approccio Code-First e attivare le Migrations

**OPZIONALE**: Implementare una o più delle funzionalità utilizzando ADO.NET (Connected o Disconnected mode, a scelta)

*Es. Inserire una nuova spesa*