# Test di fine settimana – Week 5

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Nome | Giulia |
|  |  | Cognome | Tuttobene |
|  |  | Data | 18-06-2021 |

Leggete attentamente ogni domanda e argomentare quanto più possibile **fornendo anche degli esempi**.  
ATTENZIONE: Le domande a risposta multipla possono contenere più risposte corrette.

1. *Spiegare come funziona una connection pool.*

Si tratta dell’insieme delle connessioni richiedibili dall’utente. Quando viene richiesta una connessione il pooler la cerca tra le disponibili e una volta chiusa la etichetta come nuovamente disponibile.

1. *Quale/i delle seguenti affermazioni è esatta riguardo la connection string?*

La connection string è un oggetto che gestisce la connessione a un database

La connection string è una stringa che fornisce i parametri di inizializzazione per creare una connessione

La connection string è composta da valori in formato chiave-valore

Se un utente accede al db da diverse postazioni, la connection string cambia

1. *Spiegare le differenze tra le modalità di connessione al database*

* Connected mode: connessione che permette di agire direttamente sul database. Viene definita di sola lettura poiché una volta inviato il comando al DB si può solamente “leggere” la risposta del DB.
* Disconnected mode: connessione che permette di lavorare sul DB in locale, prima acquisendo i dati e poi riconciliandoli con quelli di origine, attraverso l’utilizzo dell’adapter e del dataset.

1. *Spiegare cosa è un DataAdapter e specificare in quale modalità si usa*

L’adapter si usa in modalità disconnessa: è un ponte tra l’origine dei dati e il dataset. Gli vengono attribuiti i comandi che dovrà eseguire sui dati e si occupa della riconciliazione tra quelli di origine e quelli in locale.

1. *Data un’entità Persona con Nome, Cognome, Età scrivere una query Linq con query syntax e lambda expression che prelevi le persone maggiorenni ordinate per cognome e per nome*

CON QUERY SYNTAX:

IEnumerable<Persona> maggiorenniCognomeNome =

from m in GetAll()

where m.Età > 18

orderby m.Cognome, m.Nome

select m;

return maggiorenniCognomeNome.ToList();

CON LAMBDA EXPRESSION

IEnumerable<Persona> maggiorenniCognomeNome = GetAll().

Where(m => m.Età >18).OrderBy(m = > m.Cognome).ThenBy(m => m.Nome)

return maggiorenniCognomeNome.ToList();

1. *Spiegare la differenza tra concorrenza ottimistica e pessimistica nell’accesso ad un database.*

Sono entrambe modalità che si occupano della gestione della concorrenza. Dati due utenti concorrenti:

* La concorrenza pessimistica blocca al secondo utente (in ordine di arrivo) l’accesso alle tabelle già in uno dal primo utente
* La concorrenza ottimistica permette l’accesso a tutti gli utenti ma prima di una modifica si accerta che non ve ne siano state altre concorrenti. In questo caso si può mostrare la data di ultima modifica o il valore del record attuale che si vuole modificare.

*Esercitazione Pratica*

*Creare una Console App che gestisca la registrazione degli esami di uno Studente.*

*Lo studente è definito con:*

* *Nome*
* *Cognome*
* *AnnoNascita*

*L’esame è definito da:*

* *Nome*
* *CFU*
* *Data*
* *Votazione*
* *Passato*

*Vedere tutti gli studenti del database*

*Registrare un esame per uno studente specifico*

*Mostrare gli esami di uno studente ordinati per votazione e per data”*

*Aggiungere un nuovo studente utilizzando la modalità disconnessa di ADO.NET*

*Requisiti Tecnici:*

* *Usare la tecnologia ADO.NET*
* *Utilizzare opportunamente il Repository Pattern*
* *Realizzare le interrogazioni attraverso System.Linq in modalità Lambda Expression*

*Opzionale: Gestire l’interazione con l’utente con un menù.*

*Caricare la prova pratica e teorica su Github.*