

Programma corso .NET – Azure 2023/2024

C# PRINCIPI FONDAMENTALI, ESERCITAZIONI SU STRUTTURE DATI, CREAZIONE DI APPLICAZIONI.

- Introduzione al linguaggio C# ed all'utilizzo di Visual Studio 2022.
- La Console application in Visual Studio 2022.
- Descrivere la sintassi di base e le caratteristiche di C#
- Creare classi, definire ed implementare le interfacce, creare e utilizzare collezioni generiche.
- Creare e chiamare i metodi, catturare e gestire le eccezioni e descrivere le metodologie di logging
- Utilizzare l'ereditarietà per creare una gerarchia di classi, estendere il .NET Framework e creare classi e metodi personalizzati.
- Creazione di windows form applicazione con Visual Studio 2022.

C# AVANZATO, ESERCITAZIONI SU STRUTTURE DATI, CREAZIONE DI APPLICAZIONI.

- Leggere e scrivere dati utilizzando file di input/output e classi di Stream;
- Serializzare e deserializzare i dati in diversi formati (JSON in primis)
- Utilizzare LINQ to object per eseguire query su collections .NET.
- Introduzione a LINQ to entities per eseguire query su DB.
- Introduzione a gestioni dati complessi: HashTable, Dictionary.

PATTERN MVC .NET 7 PER LO SVILUPPO DI UN'APPLICAZIONE API REST, PATTERN ARCHITETTURALI: ENTITY FRAMEWORK, DEPENDENCY INJECTION. (NO ASP.NET MVC CORE)

- Installazione sui client dei discenti di Microsoft Information Services.
- Introduzione al framework NET 7.0
- Cosa è il pattern MVC e come è implementato da Microsoft: web applications in Visual Studio 2022: MVC .NET 7.0.
- Sviluppo di Controllers e Actions.
- Dependency Injection di servizi nei Controllers: configurazione ed uso.
- Sviluppo di Views html in MVC: HTML helpers e Tag helpers
- Sviluppo di Models in MVC: validazione ed attributi
- Configurazione di Entity Framework 7.0 in un'applicazione MVC
- Utilizzo di layouts, Javascript/Jquery e CSS nelle Views MVC
- Gestione della sicurezza: Microsoft Identity, Forms authentication e cenni a JWT

- Implementazione di JSON Rest API usando MVC .NET 7.0
- Chiamare API Rest usando C# e clients Javascript.

SVILUPPO CLIENT SIDE PER LA REALIZZAZIONE DI UN'APPLICAZIONE WEB. HTML CSS, HTML5 CSS3 JAVASCRIPT BOOTSTRAP CON REACT

Introduzione a HTML CSS, HTML5 CSS3 JAVASCRIPT BOOTSTRAP con l'approccio SPA di REACT. Concetti di componenti, services, moduli di REACT e differenze tra Javascript e Typescript.

1. Describing the UI
 - a. Writing Markup with JSX
 - b. Passing Props to a component
 - c. Conditional Rendering
 - d. Rendering Lists
2. Adding Interactivity
 - a. Responding to Events
 - b. State
 - c. Updating Objects in State
 - d. Updating Arrays in State
3. Managing State
 - a. Reacting to input with State
 - b. Choosing the State Structure
 - c. Sharing State between Components
 - d. Preserving and resetting State
4. Escape Hatches
 - a. Referencing Values with refs
 - b. Manipulating the DOM with Refs
 - c. Synchronizing with effects
 - d. Lifecycle of Reactive Effects
 - e. Separating Events from effects
 - f. Removing Effects Dependencies
 - g. Reusing Logic with Custom Hooks
5. Using Axios to manage request and response call

INTRODUZIONE E TEORIA DEI DATABASE RELAZIONALI, COME INTERROGARE DATABASE MICROSOFT SQL SERVER CON IL LINGUAGGIO T-SQL.

- Installazione sui client dei discenti di Microsoft SQL Express.
- Le basi di SQL e di Transact-SQL
- Interrogare basi di dati
- Filtrare ed ordinare i dati
- Aggregare i dati e le funzioni di aggregazione

- Query su tabelle multiple usando le clausole di JOIN
- Modificare dati: INSERT, UPDATE e DELETE.
- Le subquery in SQL
- Table expressions; CTE, tabelle temporanee, variabili tabella
- Le viste e le table-valued functions
- Stored procedures e transazioni
- Gestione degli errori

SVILUPPO CLIENT SIDE PER LA REALIZZAZIONE DI UN'APPLICAZIONE WEB. HTML CSS, HTML5 CSS3 JAVASCRIPT BOOTSTRAP CON REACT

Introduzione a HTML CSS, HTML5 CSS3 JAVASCRIPT BOOTSTRAP con l'approccio SPA di REACT. Concetti di componenti, services, moduli di REACT e differenze tra Javascript e Typescript. Implementazione Rest API e gestione dati in formato JSON (CRUD)

Produzione di una web applicazione scritta in React (client-side) che gestisce:

1. Request and response to REST API before developed.
2. Master – Details
3. CRUD OPERATION

AZURE OVERVIEW

- Introduzione al cloud, al portale Azure e ai principali servizi offerti (PAAS, IAAS e SAAS)
- Introduzione ai principali pattern architetturali in Azure
- La sicurezza in Azure – RBAC – Meccanismo di Autenticazione OAUTH2 SAML2.0
- Azure compute solutions (VM, App service Plan e serverless)
- Azure Monitor Service
- Azure Resource Manager

AZURE ADVANCED

Creazione delle risorse che serviranno per pubblicare la Rest API sul cloud Azure

- Azure Storage: Blob, Queue, Table (DB NOSQL)
- Database relazionali, SQL SERVER INSTANCE
 1. KEY VAULT

Integrazione nell' applicazione web dei seguenti punti:

1. Key vault per salvare in maniera criptata la stringa di connessione a SQL SERVER pubblicato in AZURE
2. Salva un file su un Blob Container (cartella)
3. Pubblicare l'applicazione web ed il DB su App Service Azure.
4. Registrare l'applicazione su App Registrations per implementare la login su Azure AD (Autenticazione OAUTH2 SAML2.0)

SVILUPPO APP CON REACT NATIVE - (IOS E ANDROID)

Installazione prerequisiti software ed installazione EXPO o Android Studio come emulatori dei cellulari IOS e Android.

1. Differenze fra React native rispetto a React
2. Overview about the new Architecture
3. Installazione ambiente emulatore cellulare
4. Creazione prima app con la gestione di alcuni controlli base (dropdown)
5. Salvataggio codice su github
6. Navigation e Routing
7. Gestures
8. Styling dei componenti
9. Animazioni
10. Call Rest API with Axios