formatif web3:

Quelle est la plateforme de développement Web de Microsoft?:

ASP.NET Core

Donnez une brève histoire de l'évolution d'ASP.NET jusqu'à sa version actuelle, ASP.NET Core 8:  
ASP.NET a été introduit en 2002 et a évolué pour devenir ASP.NET Core 8. Il a subi plusieurs réinventions pour rester une plateforme de développement Web moderne.

Quels sont les composants principaux d'ASP.NET Core?:  
Frameworks d'application

En quoi les frameworks d'application d'ASP.NET Core sont-ils complémentaires et en quoi résolvent-ils des problèmes différents?

Les frameworks d'application d'ASP.NET Core sont complémentaires car ils résolvent des problèmes spécifiques ou offrent des approches différentes pour les mêmes problèmes, permettant une plus grande flexibilité dans le développement.

Pourquoi est-il important de comprendre la relation entre ces frameworks?

Pour comprendre les modèles de conception pris en charge par Microsoft

Pourquoi le framework MVC a-t-il été introduit au début d'ASP.NET?   
Le framework MVC a été introduit pour offrir une meilleure façon de construire des applications web, en adoptant une approche Model-View-Controller pour une séparation claire des différentes parties de l'application.

Comment le framework MVC d'ASP.NET Core diffère-t-il de ses versions précédentes?

Le framework MVC d'ASP.NET Core a été reconstruit de manière ouverte et adaptable par rapport aux versions précédentes conçues pour les Pages Web.

Quel problème les Pages Razor tentent-elles de résoudre par rapport au framework MVC?

Travail préparatoire excessif pour générer du contenu

Comment les Pages Razor utilisent-elles les fonctionnalités initialement conçues pour le framework MVC?

Les Pages Razor appliquent la philosophie des pages Web en utilisant les fonctionnalités du framework MVC pour fusionner le code et le contenu.

Peuvent-elles être utilisées en parallèle avec le framework MVC? Oui

Quelles sont les deux versions de Blazor, et quelles sont leurs différences fondamentales?

Blazor Server et Blazor WebAssembly

Pourquoi Blazor a-t-il été développé, et quel problème vise-t-il à résoudre?

Blazor a été développé pour permettre l'utilisation de C# pour créer des applications côté client, résolvant ainsi le défi d'apprendre un nouveau langage pour le développement côté client.

Comment Blazor permet-il aux développeurs d'utiliser C# pour créer des applications côté client?

Blazor permet cela en utilisant Blazor Server, qui maintient une connexion persistante avec le serveur pour exécuter le code C# de l'application.

Quels sont les rôles respectifs d'Entity Framework Core et d'ASP.NET Core Identity dans les applications ASP.NET Core?

Entity Framework Core est un ORM utilisé pour accéder aux bases de données, tandis qu'ASP.NET Core Identity est utilisé pour l'authentification et l'autorisation des utilisateurs.

Comment Entity Framework Core est-il utilisé dans les applications ASP.NET Core?

Pour accéder aux bases de données

Quelles sont les fonctionnalités de bas niveau fournies par la plateforme ASP.NET Core?

Serveur HTTP intégré, système de composants middleware

Expliquez le rôle du serveur HTTP intégré et du système de composants middleware dans ASP.NET Core.

Le serveur HTTP intégré gère les requêtes HTTP, tandis que le système de composants middleware gère les requêtes en fournissant des fonctionnalités de base comme le routage d'URL et le moteur d'affichage Razor.

Quel est le rôle de SignalR dans les applications web?

Faciliter les communications en temps réel

Comment SignalR facilite-t-il les communications en temps réel?

SignalR établit des connexions persistantes et bidirectionnelles entre les navigateurs des utilisateurs et le serveur, permettant des mises à jour en temps réel.

Quels sont les trois points majeurs autour desquels s'articule ASP.NET Core Identity?

Couche d'abstraction, Mécanismes prêts à l'emploi, Authentification externe

En quoi ASP.NET Core Identity facilite-t-il le travail du développeur dans la gestion de l'authentification?

ASP.NET Core Identity facilite le travail du développeur en fournissant des mécanismes prêts à l'emploi et simples, tels que le login utilisateur, l'envoi d'e-mails de validation, l'inscription au site, la vérification par SMS, etc., pour simplifier la gestion de l'authentification

Quel est le rôle principal d'ASP.NET Core Identity par rapport au stockage des utilisateurs?

Abstraction entre l'application et le stockage des utilisateurs

À quelle version ASP.NET Core Identity en est-elle actuellement?

version 2.1 de ASP.NET Core

Pourquoi ASP.NET Core Identity a-t-il été complètement repensé dans sa troisième version?

Offrir une ouverture inégalée

Quel est l'objectif principal d'offrir une ouverture inégalée dans ASP.NET Core Identity?

L'objectif principal est d'ouvrir à des systèmes d'authentification externes comme Facebook ou Google, élargissant ainsi les options d'authentification pour les utilisateurs.

Pouvez-vous citer quelques-unes des fonctionnalités d'ASP.NET Core Identity en termes de mécanismes d'authentification?

Gestion des tâches asynchrones,

Gestion des tâches asynchrones,

Calcul des ressources CPU,

Routage d'URL,

Login utilisateur

Comment ASP.NET Core Identity contribue-t-il à simplifier la gestion de l'authentification?

En fournissant des mécanismes prêts à l'emploi et simples, ASP.NET Core Identity allège le travail du développeur dans la gestion de l'authentification, offrant ainsi une solution complète et robuste.

Pourquoi est-il crucial pour Identity d'être l'interface privilégiée entre le système d'information et l'authentification des utilisateurs?

Gestion centralisée de l'authentification,

Utilisation de JavaScript pour l'authentification

Quelle est la principale fonctionnalité d'ASP.NET SignalR?

Interaction en temps réel entre le client et le serveur

En quoi consiste l'interaction en temps réel entre le client et le serveur dans le contexte d'ASP.NET SignalR ?

L'interaction en temps réel signifie que le serveur peut communiquer avec les clients et envoyer des informations de manière instantanée sans rafraîchissement de la page.

Qu'est-ce qui différencie SignalR des technologies traditionnelles?

Capacité du serveur à communiquer instantanément avec les clients

Qu'est-ce qu'un RPC (Remote Procedure Call) dans le contexte de SignalR?

Un RPC dans SignalR signifie que le code .NET peut appeler du code JavaScript, et vice versa, de manière transparente pour le développeur.

Quel est le rôle d'un Hub dans SignalR?

Recevoir les appels des clients et les traiter

Expliquez comment SignalR permet de faire des broadcasts de messages.

SignalR permet de faire des broadcasts en envoyant le même message à tous les clients connectés, ou alors le serveur peut gérer lui-même à quel client il veut envoyer le message.

Comment fonctionnent les hubs dans SignalR?

Lorsqu'un hub appelle une méthode vers le client, un paquet est envoyé à travers le canal de communication utilisé

Quels sont les éléments contenus dans le paquet envoyé lorsqu'un hub appelle une méthode vers le client?

Le paquet contient le nom de la méthode et les paramètres de la méthode, sérialisés en JSON.

Quel est le rôle du Hub Proxy généré pour le client dans SignalR?

* Permettre les appels en JavaScript du client vers le serveur

Comment fonctionne le processus d'appel d'une fonction serveur depuis le client dans SignalR?

Le Hub Proxy permet au client de faire les appels en JavaScript vers le serveur, facilitant ainsi l'appel des fonctions serveur depuis le client.

Où doit être définie une classe de hub dans une application ASP.NET utilisant SignalR?

Espace de noms Microsoft.AspNetCore.SignalR

Quelle est l'utilité de la méthode BroadcastMessage dans un hub SignalR?

La méthode BroadcastMessage peut être appelée par les clients connectés au hub pour envoyer et recevoir des messages sous forme de JSON.

Quelle est la propriété de la classe Hub permettant de gérer les détails des clients connectés?

**Clients.All**

Quelle est la différence entre Clients.All et Clients.Others lors de la diffusion d'un message dans SignalR

Clients.All envoie le message à tous les clients, tandis que Clients.Others exclut le client expéditeur de la liste des destinataires.

Comment peut-on utiliser la propriété Caller pour personnaliser l'envoi de messages dans SignalR?

Renvoyer des messages uniquement à l'expéditeur

Comment peut-on utiliser Context.ConnectionId pour envoyer des messages à des clients spécifiques dans SignalR?

En utilisant Context.ConnectionId, on peut obtenir l'identifiant unique de la connexion client actuelle et envoyer des messages spécifiquement à ce client.

En quoi les groupes sont-ils utiles dans SignalR?

Les groupes sont utiles pour attribuer tous les clients d'une catégorie spécifique à un groupe, ou pour associer un groupe à un utilisateur spécifique, facilitant le suivi du client connecté qui représente l'utilisateur.

Quel est l'avantage d'utiliser des groupes par rapport aux identifiants de connexion dans SignalR?

Les noms de groupe peuvent être facilement lisibles et gérables