

- 1-)C# programlama dili nedir? Hangi alanlarda kullanılır?
- 2-)Bir programın çalışması için temel bileşenler nelerdir? (Derleyici, RAM, CPU vb.)
- 3-)Kullanıcıdan adını alıp ekrana yazdıran basit bir C# programı yazın.

Not: Kodlarınızı githuba pushlayıp bizimle paylaşınız.

1-)C# programlama dili nedir? Hangi alanlarda kullanılır?

C# Programlama Dili Nedir?

C# (C-Sharp), Microsoft tarafından geliştirilen, **.NET platformu** üzerinde çalışan, **modern, nesne yönelimli (OOP)** bir programlama dilidir. **Java ve C++** gibi dillerden esinlenerek oluşturulmuştur ve **güçlü bir tip sistemine** sahiptir.

C# dili, özellikle Windows tabanlı uygulamalar geliştirmek için optimize edilmiştir, ancak **.NET Core** ile birlikte **platform bağımsız** hale gelmiştir. Günümüzde **web, masaüstü, mobil ve oyun geliştirme** gibi birçok farklı alanda kullanılmaktadır.

C# Hangi Alanlarda Kullanılır?

1. Masaüstü Uygulamaları

- **Windows Forms (WinForms)** ve **Windows Presentation Foundation (WPF)** kullanılarak **grafik arayüzlü masaüstü uygulamaları** geliştirilir.
- Örneğin: Muhasebe, stok takip, CRM yazılımları.

2. Web Uygulamaları

- **ASP.NET Core ve ASP.NET MVC** ile **dinamik web siteleri ve web API'leri** oluşturulabilir.
- Örneğin: E-ticaret siteleri, bloglar, iş yönetim panelleri.

3. Oyun Geliştirme

- **Unity oyun motoru** C# dili ile çalışır, bu yüzden 2D ve 3D oyun geliştirmede yaygın olarak kullanılır.
- Örneğin: **Temple Run, Hollow Knight, Cities: Skylines** gibi oyunlar.

4. Mobil Uygulamalar

- **Xamarin** ile hem **Android hem de iOS** için **native (yerel) mobil uygulamalar** geliştirilebilir.
- Örneğin: Sosyal medya uygulamaları, sağlık takip uygulamaları.

5. Bulut Bilişim ve Mikroservisler

- **Azure, AWS, Docker ve Kubernetes** gibi servislerle entegrasyon yapılarak **bulut tabanlı uygulamalar** geliştirilebilir.

6. Yapay Zeka ve Makine Öğrenmesi

- **ML.NET** gibi kütüphaneler ile **makine öğrenmesi (ML) ve yapay zeka (AI)** projeleri geliştirilebilir.

7. IoT (Nesnelerin İnterneti) ve Gömülü Sistemler

- **.NET NanoFramework** ile IoT cihazları ve mikrodenetleyiciler için yazılım geliştirilebilir.

8. Veritabanı Programlama

- **Entity Framework Core (EF Core)** ve **ADO.NET** gibi araçlarla **SQL Server, MySQL, PostgreSQL** gibi veritabanları ile çalışılabilir.

2-)Bir programın çalışması için temel bileşenler nelerdir? (Derleyici, RAM, CPU vb.)

1. Donanım Bileşenleri

◆ CPU (İşlemci – Central Processing Unit)

- Programın kodlarını satır satır çalıştırır.
- Hesaplamaları ve mantıksal işlemleri yürütür.

◆ RAM (Bellek – Random Access Memory)

- Çalışan programların verilerini geçici olarak saklar.
- Program kapatıldığında veriler silinir.
- Yetersiz RAM, programların yavaş çalışmasına veya çökmesine neden olabilir.

◆ Depolama (SSD/HDD)

- Program dosyalarının kalıcı olarak saklandığı yerdir.
- **SSD** kullanımı programların daha hızlı açılmasını sağlar.

◆ Ekran Kartı (GPU – Graphics Processing Unit) (Opsiyonel)

- Grafiksel işlemleri hızlandırır.
- Oyunlar, video düzenleme ve yapay zeka gibi uygulamalarda önemlidir.

◆ Giriş/Çıkış Aygıtları

- **Klavye, fare, monitör, yazıcı, ağ kartı** gibi bileşenler programla etkileşimi sağlar.

2. Yazılım Bileşenleri

◆ İşletim Sistemi (OS – Operating System)

- Programın çalışabilmesi için gerekli ortamı sağlar.
- Örnekler: **Windows, Linux, macOS**

◆ Derleyici veya Yorumlayıcı

- Kaynak kodunu **makine diline** çevirerek çalıştırılabilir hale getirir.
- **Derleyiciler (Compiler)** → C, C++, Java gibi dillerde kullanılır.
- **Yorumlayıcılar (Interpreter)** → Python, JavaScript gibi dillerde satır satır çalıştırma yapar.

◆ Çalışma Zamanı Ortamı (Runtime Environment)

- Derlenmiş kodun işletim sistemi tarafından çalıştırılmasını sağlar.
- Örnekler: **JVM (Java Virtual Machine)**, **.NET CLR (Common Language Runtime)**

◆ Bağımlılıklar ve Kütüphaneler

- Programların belirli işlevleri gerçekleştirmek için kullandığı **hazır kodlar**.
- Örnekler: **Node.js**, **.NET Framework**, **OpenGL**, **TensorFlow**

◆ API'ler (Application Programming Interface)

- Programın işletim sistemi veya diğer yazılımlarla **iletişim kurmasını** sağlar.
- Örnekler: **Windows API**, **RESTful API**, **DirectX**