İBB ECODATİON FRONT END BOOTCAMP

1- İnterpreter ve compiler nedir?

Yorumlayıcılar (Interpreter); kodu satır ve/veya bloklar halinde çalıştırırlar ve bir sonraki satır / blokları sırası gelmeden değerlendirmezler. Bu nedenle, sonraki satırdaki hatalar ve kodun bütününü etkileyen iyileştirmeler yorumlayıcılar tarafından yakalanamazlar. Kodu parçalar halinde değerlendirmek amacıyla kullanılırlar.

Bazı programlama dilleri derleyici ve yorumlayıcılara sahip olabilir. Ancak, daha kapsamlı çözümler sunabilen harici derleyici ve yorumlayıcılar da çalıştırılabilmektedir.

Derleyici (Compiler); Bir dilde yazılmış olan kodu (kaynak kodu / source code) başka bir dilde yazılmış koda, örneğin yüksek seviye bir dilden (high-level programming language) alt seviye bir dile (low-level programming language), dönüştüren araçlardır. Genellikle bu kodlar çalıştırılabilir (executable) şekilde üretilirler. Bunun nedeni, çoğunlukla diller arasındaki bu işlemin pratik bir şekilde işletim sistemi ve donanım bağımlı kodlara dönüştürülebilmesidir. Örneğin, C dilinde yazılmış bir kodun PASCAL koduna dönüştürmesi gibi.

2- Senkron ve asenkron programlama nedir?

Senkron programlamadaki her şeyi sırayla işlemesi ve her bir işlemin birbirini beklemesi yeri geldiğinde programımızı çok yavaşlatabilir, hatta işlem bitene kadar durdurabilir. Örneğin yukarıdaki kodda 5. satır bir önceki satırı yani dosya okuma işlemini beklemek zorundadır. Dosya içeriği çok büyükse bu işlemler dakikalar bile alabilir. Ekrana *“Program çalışıyor…”* yazdırmak için bir önceki işlemin bitmesini beklemek pek akıllıca değildir. İşte bu tip durumlar için asenkron fonksiyonlar kullanırız. Kod akışının sırayla işlemediği, işlemlerin birbirini beklemediği, kod akışının işlem durumlarına göre devam ettiği programlamaya Asenkron Programlama denir.

Asenkron(Async) programlamaya neden ihtiyaç duyulmuş ilk olarak bu soruyu cevaplayalım. Kullanmakta olduğumuz programlarda aynı anda birden çok işlem yapılabilmektedir. Örneğin siz kullanıcı arayüzünde bir yazı görüyorken arka planda bir web servisine istek gönderilmiş ve cevabı bekleniyor olabilir. İlk nesil programlarda web servisinden cevap gelene kadar siz kullanıcı arayüzünde herhangi bir tuşa basamazdınız. Eğer basarsanız bir şey olmayacaktır ve peş peşe bir kaç defa basarsanız program durduruldu hatası alırdık. Asenkron fonksiyon tanımlamak özellikler farklı tipte işlemleri bir arada yürütüyorken kullanmak oldukça pratik ve sağlıklıdır. Ancak her fonksiyonu Asenkron olarak yazılmaz. Dataların kontrolünü kaybedebilirsiniz.

Kısacası asenkron programlama programın senkron bir şekilde değil de öncelik verdiğimiz işlemlerin daha önce yapılmasını sağlayan ya da sağladığımız programlama türüdür.

3- DOM (Document Object Model) nedir?

DOM, belgenin içeriğini belirli bir düzen içerisinde nesne olarak temsil edilmesini sağlayan bir modeldir. Bu modelde belge yapısında bulunan her bir nesne için bir düğüm bulunur. Alt düğümler kök bir düğüme bağlanır. Belge ile iletişim kuran servis düğümü çözerek işlemlerini gerçekleştirir. Örneğin, bir WEB [API](https://www.hosting.com.tr/bilgi-bankasi/api/) servisi, XML dosyadan veri çekerken bu modele göre belgeye erişir.

İnternet tarayıcısı bir web sayfasını açarken sunucuya ilk isteği gönderir ve sayfaya ait belge yükler. Belgedeki nesneleri düğümlerden çözerek hiyerarşiye göre sayfa içerisine yerleştirir ve site açılmış olur. Web sayfasına dışardan erişim sağlayan tüm servisler bu şekilde kolaylıkla belgenin mantıksal yapısına ulaşır ve sayfada değişiklik yapabilir.

4- Number( ) parseInt( ) nedir aralarındaki farklar nelerdir?

Number( ) ve parseInt( ) JavaScript programlama dilinde kullanılan iki farklı yöntemdir ve her ikisi de bir değeri sayıya dönüştürmek amacıyla kullanılır. Ancak aralarında bazı farklılıklar vardır.

Number( ) fonksiyonu, verilen değeri sayıya dönüştürmeye çalışır. Eğer dönüştürme başarılı olursa, sayıyı döndürür. Eğer dönüştürme işlemi başarısız olursa (örneğin, verilen değer bir metin içeriyorsa), NaN (Not a Number) değerini döndürür. Number() fonksiyonu genellikle sayısal dönüşümlerde tercih edilen bir yöntemdir.

parseInt( ) fonksiyonu ise verilen bir metni tam sayıya dönüştürmeye çalışır. Fonksiyon, metnin başlangıcından itibaren sayısal karakterler okur ve bu karakterleri bir tam sayıya dönüştürür. Dönüşüm işlemi metnin ilk karakteri bir sayısal karakter oluncaya kadar devam eder. Bu nedenle, parseInt( ) fonksiyonu genellikle metin içerisindeki başlangıçtaki sayısal değerleri almak için kullanılır.

5- String( ) data.toString( ) nedir aralarındaki farklar nelerdir?

String( ) ve data.toString( ) JavaScript programlama dilinde kullanılan iki farklı yöntemdir ve her ikisi de bir değeri metin formatına dönüştürmek amacıyla kullanılır. Ancak aralarında bazı farklılıklar vardır.

String( ) fonksiyonu, verilen bir değeri metin formatına dönüştürmeye çalışır. Bu fonksiyon, herhangi bir veri türünü (sayı, boolean, dizi, nesne, vb.) alabilir ve onu eşdeğer bir metin temsiline çevirir.

data.toString( ) ise bir nesnenin toString( ) metodunu çağırarak o nesneyi metin formatına dönüştürmeye çalışır. toString( ) metodunun tanımlı olması durumunda, nesnenin özel bir şekilde metin temsiline dönüştürülmesi sağlanır. Bu yöntem, genellikle nesnelerin özel bir temsilini veya özelleştirilmiş bir metin çıktısını elde etmek için kullanılır.

6- Binary, octal, decimal, hexadecimal nedir?

Binary, octal, decimal ve hexadecimal, sayı sistemleridir. Her biri farklı bir temel kullanarak sayıları temsil etmek için kullanılır.

Binary (İkili) Sistem: İkili sistem, yalnızca 0 ve 1 rakamlarını kullanır. Bilgisayarların temelinde kullanılan bir sayı sistemidir. Her bir basamağı 2'nin üssü olarak değerlendirir. Örneğin, "1010" ikili sistemde onlu karşılığı 10'dur.

Octal (Sekizli) Sistem: Sekizli sistem, 0-7 arasındaki rakamları kullanır. Her bir basamağı 8'in üssü olarak değerlendirir. Örneğin, "27" sekizli sistemde onlu karşılığı 23'tür.

Decimal (Onlu) Sistem: Onlu sistem, günlük yaşamda yaygın olarak kullandığımız sistemdir. 0-9 arasındaki rakamları kullanır. Her bir basamağı 10'un üssü olarak değerlendirir. Örneğin, "123" onlu sistemde kendisiyle aynı değeri temsil eder.

Hexadecimal (Onaltılı) Sistem: Onaltılı sistem, 0-9 rakamlarının yanı sıra A, B, C, D, E ve F harflerini kullanır. Her bir basamağı 16'nın üssü olarak değerlendirir. Bu sistemde 10'dan 15'e kadar olan sayılar, A'dan F'ye kadar olan harflerle temsil edilir. Örneğin, "1A" onaltılı sistemde onlu karşılığı 26'dır.

6- let const nedir aralarındaki farklar nelerdir?

let ve const, JavaScript programlama dilinde değişkenlerin tanımlanması için kullanılan iki farklı anahtar kelimedir. İşlevleri ve kullanımları arasında bazı farklılıklar vardır.

let:

let ile tanımlanan bir değişken, blok kapsamında (örneğin, if ifadesi veya döngü içinde) geçerlidir.

Değişkenin değeri daha sonra değiştirilebilir (yani, yeniden atama yapılabilir).

Aynı kapsamda aynı isme sahip birden çok let değişkeni tanımlanabilir.

const:

const ile tanımlanan bir değişken, blok kapsamında geçerlidir ve sabit (değiştirilemez) değere sahiptir.

Bir const değişkeni tanımlandıktan sonra, değeri değiştirilemez. Yani, bir kez atandıktan sonra yeniden atama yapılamaz.

Aynı kapsamda aynı isme sahip başka bir const değişkeni tanımlanamaz.