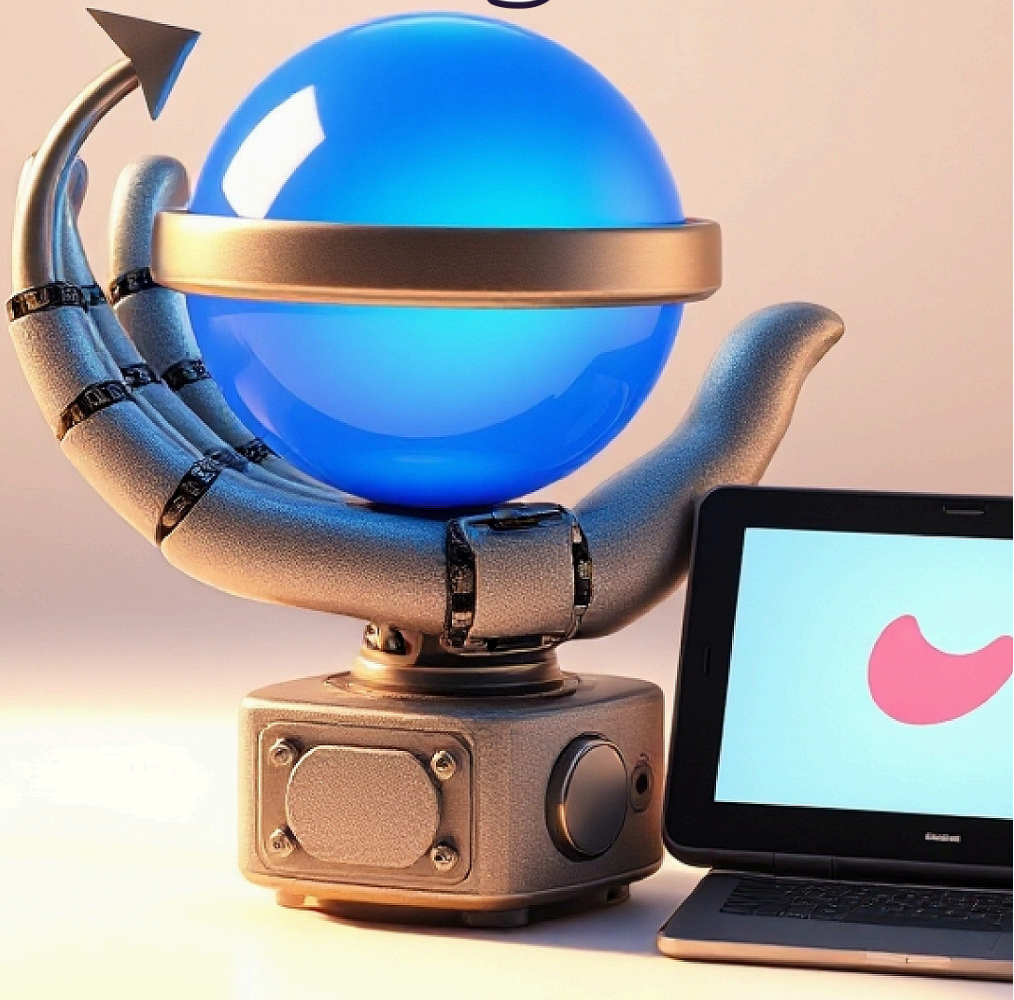
Bir fonksiyonun değer döndürmesi ne anlama gelir?



www.yzokul.com.tr

Bir fonksiyonun değer döndürmesi ne anlama gelir?



Bir fonksiyonun değer döndürmesinin ne anlama geldiğini merak edebilirsin. Kısa cevap şudur: Bir fonksiyon sonucunu, onu çağıran yere geri verebilir. O yere çağıran denir. Fonksiyon işini yapar ve sonra sonucu çağırana verir. Bu geri verme işlemi, değer döndürmenin anlamıdır.



Fonksiyon, tek bir iş yapan, adlandırılmış bir talimat bloğudur. Fonksiyona bir ad verirsin ki onu daha sonra kullanabilesin. O işin yapılmasını istediğinde fonksiyonu adıyla çağırırsın. Fonksiyon ardından talimatlarını çalıştırır. Bunu, ihtiyaç duyduğunda kullanabileceğin bir araç olarak düşün.



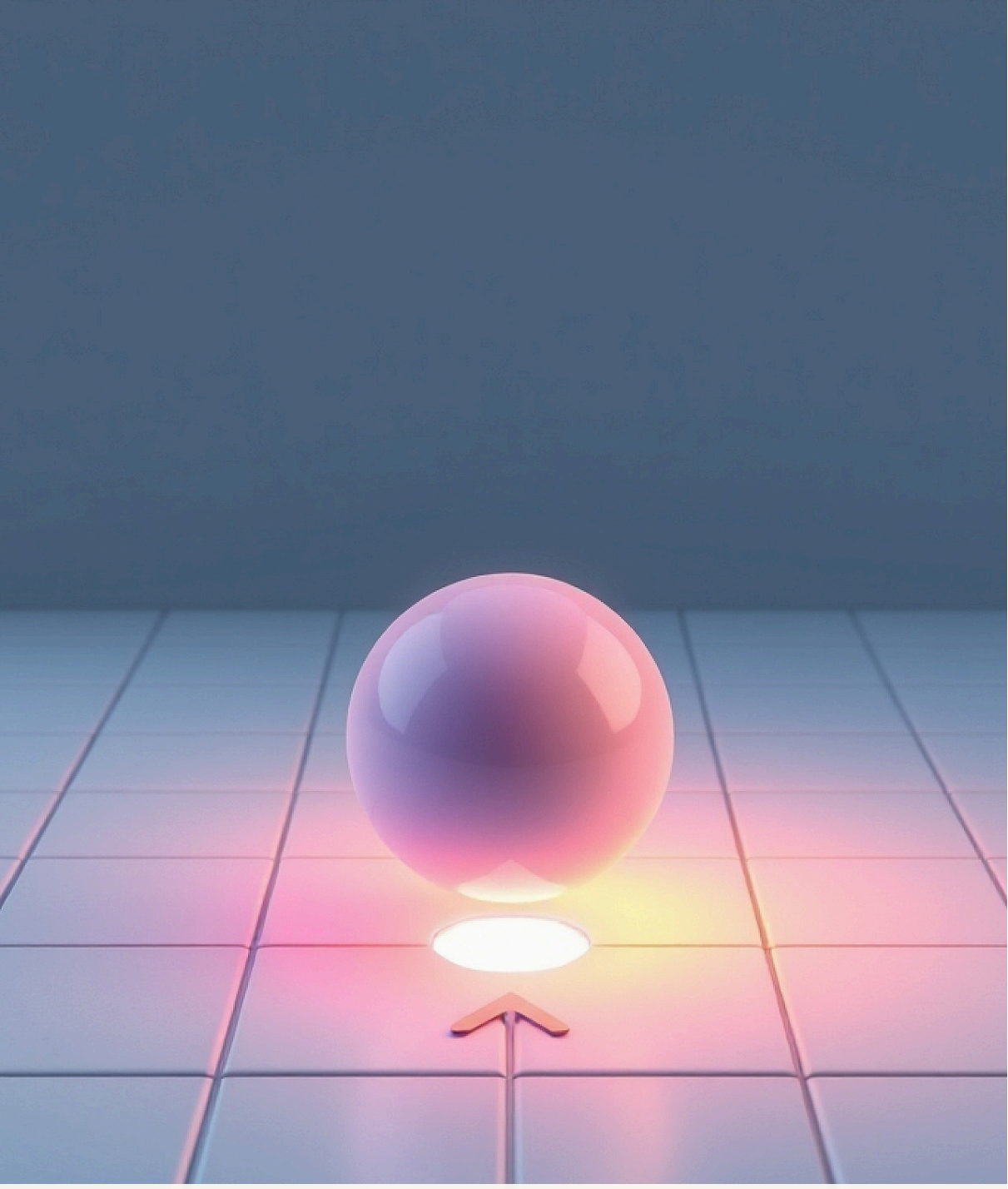
Kodunda fonksiyonu çağıran yere çağıran denir. Çağıran, fonksiyonun adını yazarak ondan işini yapmasını ister. Örneğin, add adında bir fonksiyonun varsa, onu add yazarak çağırırsın. Çağıran, yalnızca çağrıyı yapan koddur. Fonksiyonun bitmesini bekler.



İki sayıyı toplayan basit bir fonksiyona bak. İşi, bu sayıların toplamını üretmektir. Ona iki sayı verirsin ve toplamı hesaplar. Örneğin, ona 2 ve 3 verirsen 5 üretir. Fonksiyonun amacı bu toplamı kullanılabilir hale getirmektir. Toplam, onun işinin sonucudur.



Fonksiyon toplamı çağırana geri verir. Bu geri verme işlemi tam olarak değer döndürmenin anlamıdır. Fonksiyon toplamı kendine saklamaz; onu döndürür. Çağıran, toplamı çağrının sonucu olarak alır. Böylece değer fonksiyondan çağırana doğru gider.



Çağıranın sonucu, çağrıyı yaptığı tam noktada aldığını düşün. Fonksiyon bittiğinde değer o noktaya geri akar. Örneğin, kodunda add(2, 3) yazarsan orada 5 değeri belirir. Çağıran bu değeri hemen kullanabilir. Değer artık çağrı noktasında kullanılabilir durumdadır.



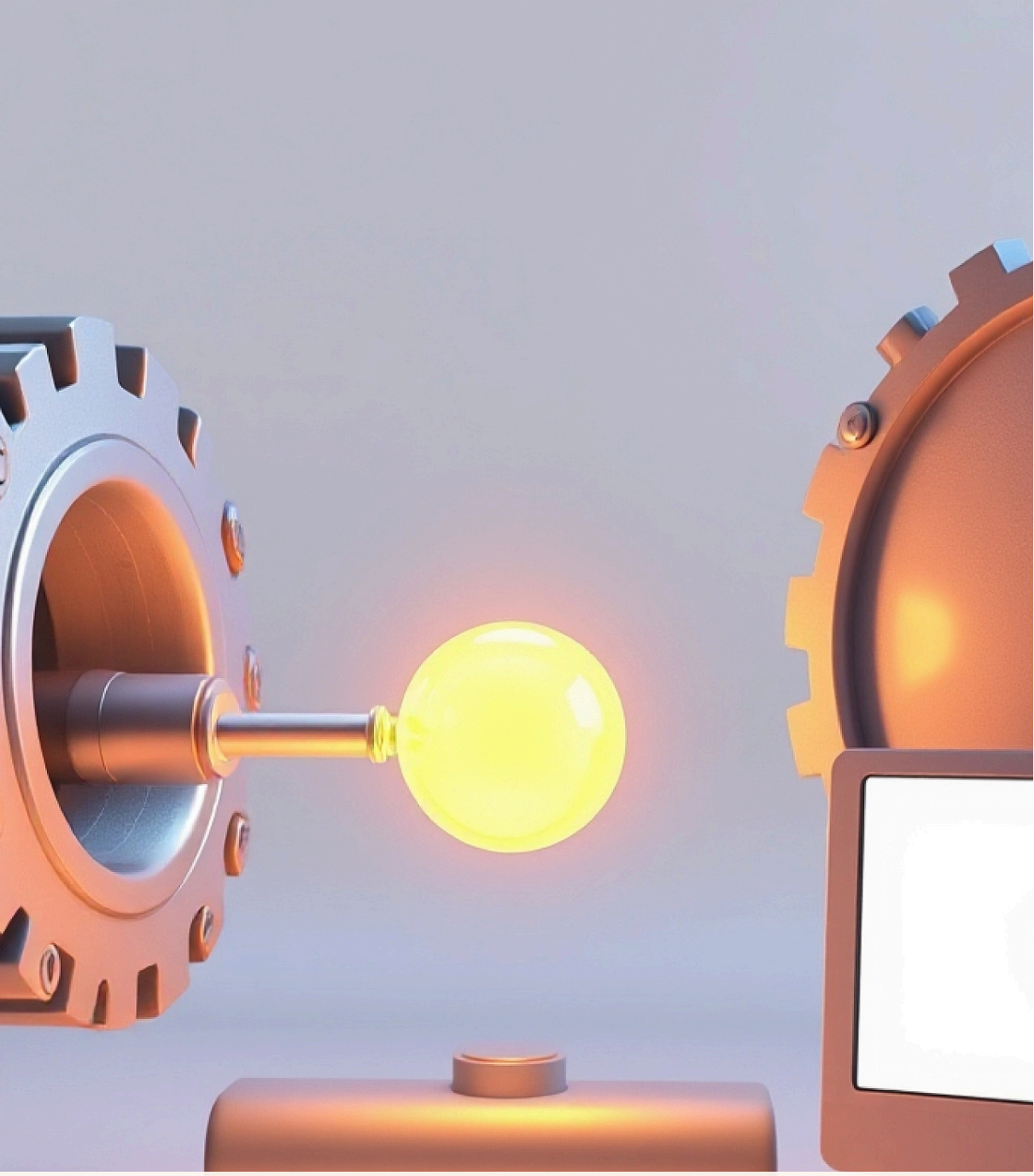
Yazdırma, bir kişinin okuması için ekranda metin gösterir. Döndürme ise sonucu daha sonra kullanılmak üzere programa geri verir. Bunlar iki farklı işlemdir. Bir fonksiyon değer döndürmeden bir şey yazdırabilir. Ayrıca hiçbir şey yazdırmadan bir değer döndürebilir. Hangisini istediğini bilmen gerekir.



Bir fonksiyon ekrana bir sayı yazdırıp yine de işe yarar bir şey döndürmeyebilir. Yazdırma yalnızca bilgi gösterir; çağırana bir sonuç geri vermez. Örneğin, bir fonksiyon toplamı yazdırabilir ama hiçbir şey döndürmeyebilir. Çağıran yazdırılan sayıyı görür ama kullanacak bir değer almaz. Yazdırma, döndürme değildir.



Döndürülen değer bir değişkene atanabilir. Değişken, bir değeri tutan adlandırılmış bir saklama yeridir. Örneğin, result = add(2, 3) yazabilirsin. result değişkeni o zaman 5 değerini tutar. Bu değişkeni programında daha sonra kullanabilirsin. Sonucu saklamak, onu ilerisi için elinde tutmanı sağlar.



Döndürülen değer doğrudan başka bir fonksiyona gönderilebilir. Bir fonksiyonun sonucu, başka bir fonksiyonun girdisi olur. Örneğin, print(add(2, 3)) yazabilirsin. Burada add fonksiyonu 5 döndürür ve bu 5 print fonksiyonuna aktarılır. Bu zincirleme, fonksiyonları kolayca birleştirmeni sağlar. Sonuç bir fonksiyondan diğerine akar.



Döndürülen değer bir koşulda kontrol edilebilir. Koşul, programın bundan sonra ne yapacağına karar veren bir testtir. Örneğin, if add(2, 3) > 4: yazıp ardından bir şey yapabilirsin. Program, döndürülen değerin 4'ten büyük olup olmadığını kontrol eder. Buna göre bir yol seçer. Bu, sonucu kullanarak karar vermeyi sağlar.



Değer döndürmek, bir fonksiyonu birçok yerde yeniden kullanılabilir kılar. Sonucuna ihtiyaç duyduğunda aynı fonksiyonu çağırabilirsin. Sonuç, gördüğün gibi saklanabilir, aktarılabilir veya kontrol edilebilir. Bu esneklik, fonksiyonları yapı taşlarına dönüştürür. Tek seferlik eylemler yerine tekrar tekrar kullanabileceğin araçlar elde edersin.



Yeni başlayanların sık yaptığı bir hata, yazdırılan bir sayının sonuç olarak kullanılabileceğini beklemektir. Bir fonksiyon yalnızca yazdırıyorsa, çağıran kullanacak bir şey geri almaz. Örneğin, add toplamı yazdırıp onu döndürmüyorsa, toplamı bir değişkende saklayamazsın. Daha sonra kullanmak için değeri döndürmen gerekir.



Bir başka hata, değer döndüren bir fonksiyonu çağırıp döndürülen değeri hiç saklamamak veya kullanmamaktır. Değer geri verilir ama sonra atılır. Örneğin, add(2, 3) yazıp bunu bir değişkene atamamak sonucu boşa harcar. Sonucu tutmak için onu bir değişkene ata veya hemen kullan. Aksi halde döndürülen değer kaybolur.



Yazdırma ve döndürme farklı amaçlara hizmet eder. Bir fonksiyon yazdırabilir, döndürebilir veya ikisini birden yapabilir. Görevin için hangisine ihtiyacın olduğunu bilmelisin. Bir sonucu bir kişiye göstermek istiyorsan yazdır. Sonucu programında kullanmak istiyorsan döndür. İkisini karıştırmak hatalara yol açar.



Değer döndürmek, sonucu çağırana geri vermek demektir. Döndürülen sonucun bir değişkende saklanabileceğini, başka bir fonksiyona aktarılabileceğini veya bir koşulda kontrol edilebileceğini unutma. Bu, yalnızca ekranda metin gösteren yazdırmayla aynı şey değildir. Fonksiyonları yazarken ve kullanırken bu fikirleri aklında tut.