**OPERATÖRLER VE MATEMATİKSEL İFADELERİN KODLAMASI**

**OPERATÖRLER**

İşlem yapmamızı sağlayan işaretlere operatör(işleç) adı verilir. Bu operatörleri aritmatiksel(+,-,\*,/), gibi mantıksal(ve,veya değil), gibi ve karşılaştırma(<,><=),gibi ve bit işlem operatöreri isimleri altında sınıflandırılırlar. Operatörlerde işlem yaparken işlem önceliği ve çalışma sırasına dikkat edilmeli. Python'da her bir operatörün bir fonksiyon eşdeğeri vardır. Örneğin "a+b" yerine "add.(a,b)" fonksiyonu kullanılabilir. Bunun için import operator programa import edilmeli.

**ARİTMATİKSEL ATAMA OPERATÖRLERİ**

Aritmatiksel ya da sözel bir ifadeyi bir değişkene aktarmak/atamak için"=" atama operatörü kullanılır. Python'da matematiksel işlemlerde; "=" atama operatörü ile aritmatiksel operatörlerin birleşmesiyle yeni operatörler oluşturulabilir ki bu yeni operatörlere aritmatiksel atama operatörleri denir.(+=,-=,\*=,/=)gibi. Örneğin += operatörünün işlevi "değişkene eşitliğin sağındaki değeri ekle ve sonucu yine değişkene aktar"dır. A +=8 A=A+8

**KARŞILAŞTIRMA OPERATÖRLERİ**

Verilerin birbiriyle karşılaştırılmasında kullanılır. Sonuç doğru ise True, yanlış ise False değerini üretir. "A=B" işlemi; "B'nin değerini A'ya aktar" anlamına gelirken "A==B" işlemi; "A değişkeninin değeri B değişkeninin değerine eşit mi ?" veya "A is B" anlamına gelmektedir.

Açıklama: min() fonksiyonu iki değeri karşılaştırır. Dolayısıyla üç değeri karşılaştırırken önce herhangi ikisi sonra diğeriyle karşlaştırılır. Örn= print(min(min(a,b),c)

**MANTIKSAL OPERATÖRLER**

Mantıksal operatörler, birden fazla verinin birbiri ile kıyaslanması durumunda kullanılır. Mantıksal operatörlerin sonucu, boolean tipi değişkenlerde tutulur ve sonuç olarak sadece birbirnin tersi olan iki değerden "True/Doğru" veya "False/Yanlış" biri üretilir.

* and Operatörü: Mantıksal(Lojik) VE (AND) işlemi operatörüdür.
* or Operatörü: Mantıksal (Lojik) VEYA (OR) işlemi operatörüdür.
* not Operatörü: Lojik DEĞİL (NOT) işlemi operatörüdür.

**BİT İŞLEM OPERATÖRLERİ**

Bit işlem operatörleri; verilen sayının/değerin ikili tabandaki bit karşılığını alıp, bitler üzerinde tek tek mantıksal (lojik) işlem gerçekleştirmektedirler. Bit işlemleri ile ilgili değişkenlerin tam sayı veri tipleri (int) olmasına dikkat edilmelidir, çünkü veri tipleri (folat,complex gibi) kullanılmamaktadır.

Bit düzeyinde bir sayının bir basamak **(bit) sola (<<)** kaydırılması o sayının **2 ile çarpılması**, bir bit sağa (>>) kaydırılması o sayının 2'ye bölnmesi anlamına gelir.

Aynı aritmatiksel atama operatörlerinde olduğu gibi, bit işlem operatörleri de atama operatörleriyle birleştirilebilir. Böylelikle karşılaştırma işlemlerinde ' >>=, <<=, &=, I=' şeklinde bitsel atama operatörleri de kullanılabilir.

**Seçimli Yapılar**

**Giriş** : Normal bir programın işleyişi, ilk satırdan son satıra doğru sırası ile adım adım gerçekleşir. Fakat bu işleyiş sırası, bir koşula bağlı olarak değiştirilebilir. Programın işleyişi sırasının bir koşula bağlı olarak değiştirilmesi işlemine **branching** (şartlı dallanma) adı verilir ve dallanma işlemini gerçekleştiriren komutlara da karar komutları denir. Karar yapılarında koşula (bir şarta) bağlı olarak iki ya da daha fazla seçenekten birini seçme işlemi gerçekleştirilmektedir. Program; şarta bağlı olarak hangi işlemi yapacağına karar verir. Python'da 'if','if else',ve 'if-elif else' olmak üzere üç adet seçme işlemini gerçekleştiren yapı vardır.

**if**: (Tek Seçimli Yapı): Koşula bağlı olarak tek bir işlemi yerine getiren yani branching işlemini gerçekleştiren yapıdır. Hemen hemen bütün programlama dillerinde bu işlem için if deyimi/komutu kullanılır. Eğer koşul kısmında birden çok fazla değerin karşılaştırılması yapılıcaksa mantıksal operatörler (and,or) kullanılabilir. if: sorgularonda koşuldan sonra ':' koymayı unutmamak ve büyük/küçük ayrımına dikkat etmek lazım.

Indentation (Kod Bloğu: Girinti Mekanizması) : Programlama dillerinde kod bloklarını belirlemek için {...} süslü parantezler ya da begin ... and sözcükleri kullanılırken Python'da ise girinti mekanizması kullanılır. Python'da dikey olarak aynı hizada başlayan ardışık kod satırları bir blok kabul edilir. Pratikre her bir blok girintisi için 4 boşluk (1 tab) bırakılması tavsiye edilir.

**if: else:** (Çift Seçimli Yapı) : Koşula bağlı olarak iki işlem yerine getiren yani branching işlemini gerçekleştiren yapıdır. Tüm programlama dillerinde olduğu gibi Python'da da bu işlem için if/else deyimi kullanılır. if: ve else:'den sonra ':' koymak unutulmamalı.

**if: elif: else:** (Çok Seçimli Yapı) : Çoklu seçimlerde if-elif deyimi merddiven basamakları şeklinde iç içe kullanılır. Koşul sayısı ikiden fazla olduğunda tercih edilir.