**Java Nedir?**

* **Nesne yönelimli programlama**

**1-** pure object oriented olma gayreti vardır,

**2-** nesne yönelimli bir dil olmanın en yılmaz savunucusudur,

**3-** herhangi bir platforma bağlı değildir,

**4-** bir kere yaz her yerde çalıştır felsefesi vardır, her marka bilgisayar ve işletim sisteminde uygulanmadan çalışır, bu çok zor bir özelliktir,

**5-** işletim sistemleri bu dili tanımaz bilmez ama JVM (Java sanal makinesi), JRE (Java çalışma ortamı-içindeki bazı kütüphaneler sayesinde çalışma ortamı oluşturur), JDK (geliştirme ortamı sunar) sayesinde sistemlerin anlayacağı dillere çevrilir.

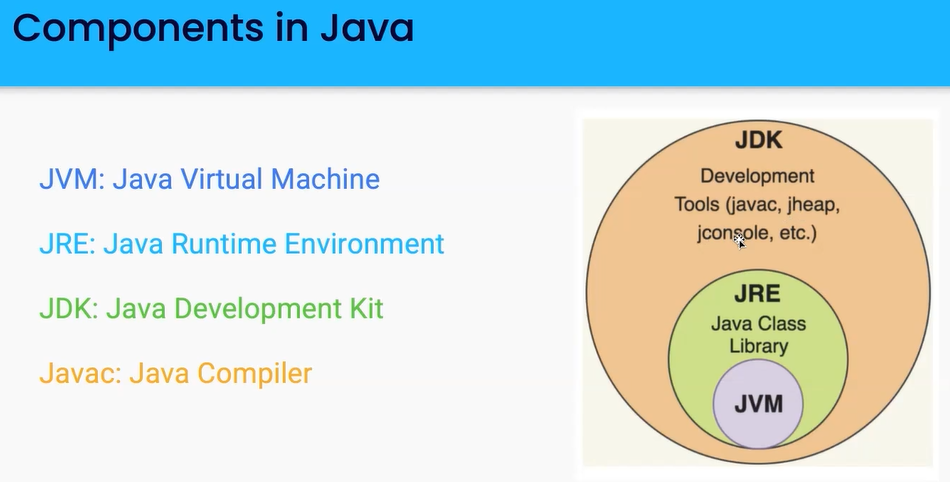
**6-** gerçek hayatı birebir kopyalar, zihin bu yüzden kolay uyum sağlar kod sistemine

**JVM =** Yazdığımız kodları sistemlerin anlayacağı Byte koda çeviren yapıdır

**JRE =** Bazı kütüphaneler + JVM

**JDK =** Bazı kütüphaneler + JRE

**6-** JDK diğer hepsini içinde barındırdığı için sadece **JDK (Java Development Kit)** kurmak yeterlidir.



* **Yapısal bir programlama dilidir** (fonksiyonları, döngüleri, koşullu ifadeleri içerir).

* **Syntax açısından C ailesine** çok yakındır. Özellikle C-Sharp a benzerdir.

* **Yapısal programlama dilinin (Structured programming language) temel 3 özelliği:**

**1- Fonksiyonlar (Metotlar)**

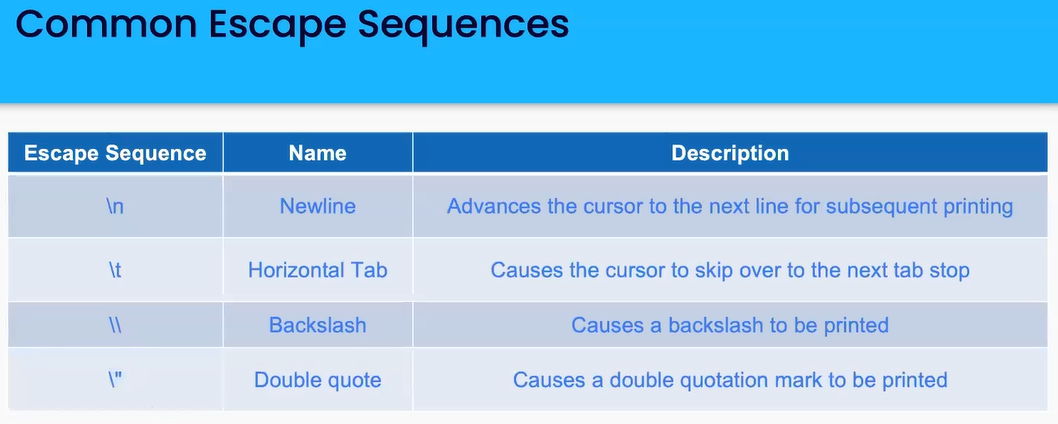
**2- Döngüler**

**3- Koşullu ifadeler (Blogların çalışıp çalışmamasının koşula bağlanması)**

**Kod bloğunun parametrize edilebilmesi gerekir. Herhangi bir kod bloğunun farklı bir parametreye göre farklı amaçlarla kullanılabilmesi.**

**Temel Kavramlar:**

* **Void:**  Ne parametre alır, ne de bir şey döndürür.
* **Paket:** Ana kümedir. Paket içinde paket de olabilir. Bilgisayardaki klasör düzeni gibidir. İç içe istediğin kadar klasör oluşturabilirsin.
* **Class:** Paketin içindeki alt kümelerdir. Paket daha düzenli olsun diye kullanılır. Paketsiz de doğrudan class oluşturulabilir.
* **Main Method:** Bir projede bir tane olması yeterlidir. Arabanın kontağı gibidir. Bu olmadan kodlar çalışmaz. Çalıştır butonuna basınca ilk aranan şey main metottur. Buna arar, bulursa kodlara geçer.
* **Println:** Tek satır. ln (line) satır başı yaptırır.
* **Print:** Ortak satır. Herşey tek satırda yazdırılır.
* **Yorum Satırı: //** yorum komutudur. **/\*...\*/** yapılırsa araya giren her şey yorum olur.
* **Satır Atlama:** Tek **System.out.println** kullanarak da satır satır kod yazdırılabilir. bunun için **ters slash n “\n”** bitişik olarak araya eklenir.
* **Ctrl + Boşluk =** Yazım kolaylığı için seçenek penceresini çıkarır.

  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
**Veri Tipleri:**

3 temel veri tipi vardır.

**1- İlkel veri tipleri:** 8 tanedir

byte (-128 ile 127 arasındaki sayılar, 8 bit yer kaplar), short (16 bit yer kaplar),

int (32 bit yer kaplar, iki milyara kadar),

long (64 bit yer kaplar),

float (32 bit yer kaplar, kısa ondalıklı gösterimler, sayının sonuna bitişik f yazılır),

double (64 bit yer kaplar, uzun ondalıklı gösterimler, sayının sonuna bitişik d yazılır), bir şey yazılmazsa bunu double olarak algılar),

char (tek karakterden oluşan verileri tutar, değeri tek tırnak işareti içerisine yazılır)

boolean(true veya false)

**2-** **Nesne (tanımlanmış, referans veri tipi)**

**3- Tipsiz (null)**

**Yazım kuralı şu şekildedir:** “veri tipi + değişkenin ismi = değer” (int sene: 2022;)

**Değişken İsmi Yazarken Dikkat Edilecek Hususlar:**

**1-** Türkçe karakter hiç kullanılmaz.

**2- camelCase Kuralı:** İlk kelimenin tümü küçük, sonrakileri bitişik ve sadece ilk harfleri büyük yazılır.

**3-** Değişken isminin başında sadece $ ve “\_” sembolü kullanılabilir. Bunun dışında hiçbir sembol kullanılmaz.

**4-** Değişken adındaki sayılar başa konulmaz. Sonda olur.

Formatting the codes (Hizalamayı otomatik yaptırmak için):

windows: ctrl + alt + L

**Operatörler:**

**1- “==” :** Eşittir sembolü. Tek eşit atama olduğu için iki tane kullanılır.

**2-** **“!=” :** Eşit değildir sembolüdür.