

Hafta-1 Java Giriş

Batuhan Düzgün - Şubat 2020

Java Nedir?

Java platform bağımsız, **Nesneye Dayalı Programlama**'yı (Object Oriented Programming) tamamiyla destekleyen bir yazılım geliştirme dilidir.

Java aynı zamanda geliştirme platformunun kendisidir.

Java Programlama Dilinin Özellikleri

Java platform bağımsız bir dildir. (Windows, Linux, Mac OS)

Java Nesneye Dayalı programlamaya tamamiyle uygun bir dildir.

Java öğrenmesi basit bir dildir.

Java kararlı bir programlama dilidir.

Java, Multi-Thread (Çok Kanallı) programlamayı varsayılan olarak destekler.

Dağıtık sistemler ve Web programlama yapabilmenize imkan tanır.

Mobil ve Gömülü sistemlerde programlama imkanı sağlar.

Java Tarihçesi

1991 yılında Green Project ile Sun Microsystem'de başlar.

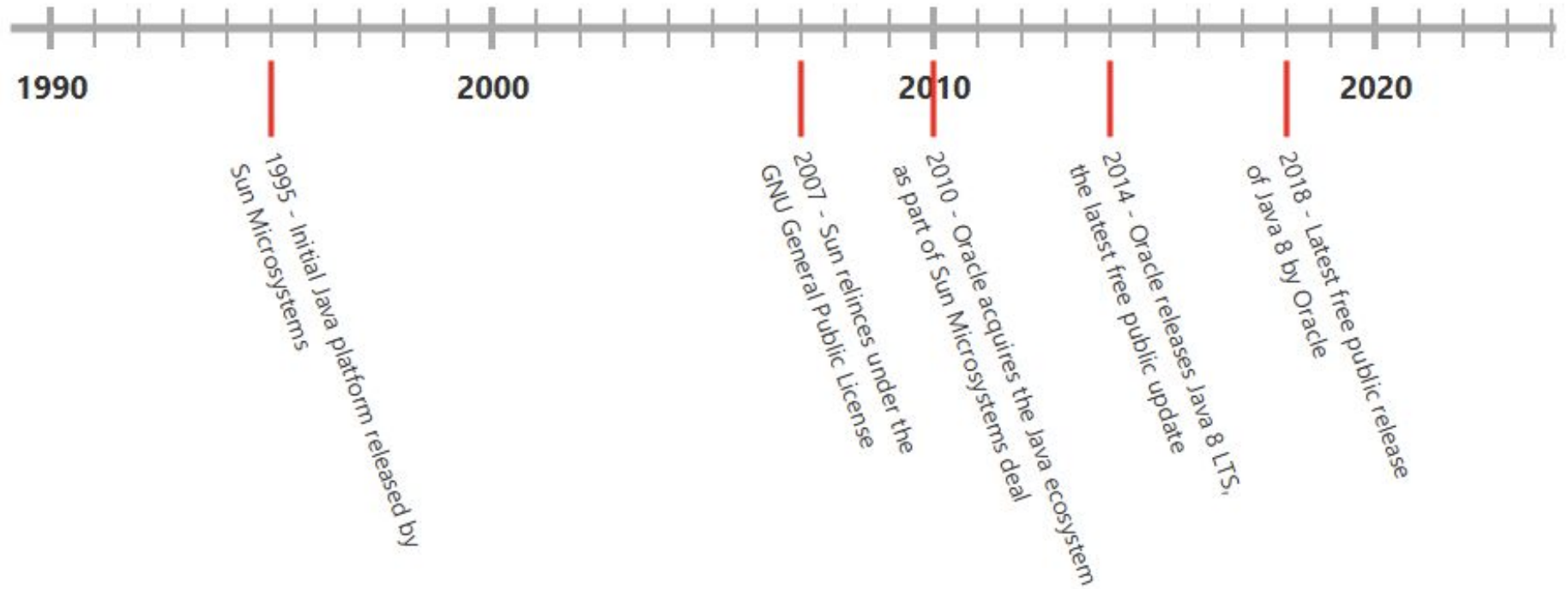
İlk ismi Oak (Yeşil Meşe Ağacı)

1995'te ilk Java Beta duyurulur.

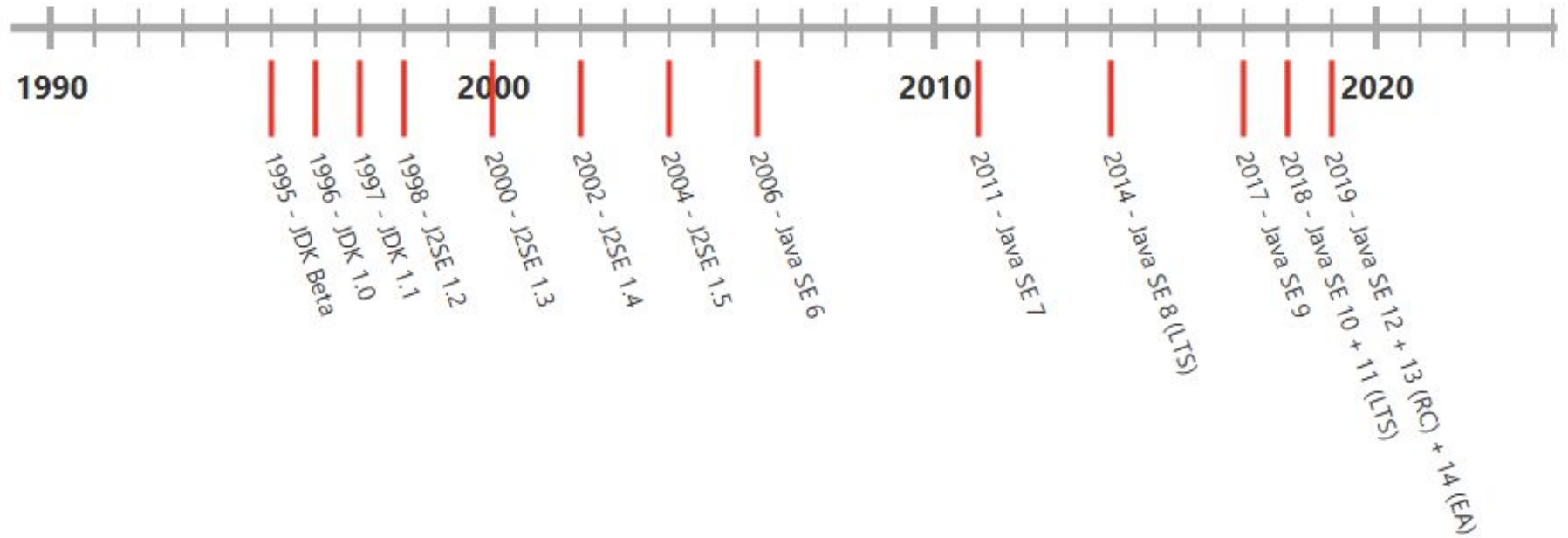
1996'da Java 1.0 adıyla resmen duyurulur.

2010 yılında Oracle firması tarafından satın alınır.

Java Tarihçesi



Java Sürüm Tarihçesi



Java ile Uygulama Geliştirme Yapabileceğiniz Alanlar

- Web uygulamaları geliştirebilirsiniz.
- Masaüstü uygulamaları geliştirebilirsiniz. (Eclipse IDE gibi)
- Kurumsal uygulamalar geliştirebilirsiniz. (www.sahibinden.com gibi)
- Mobil uygulamalar geliştirebilirsiniz.
- Gömülü sistem uygulamaları geliştirebilirsiniz.
- Robotik projelerde kullanabilirsiniz.
- Oyun programlamada kullanabilirsiniz. (Android ile oyun geliştirme gibi)

Java Uygulama Geliştirme Altyapıları

Java SE (Java Standard Edition)

Java EE (Java Enterprise Edition)

Java ME (Java Micro Edition)

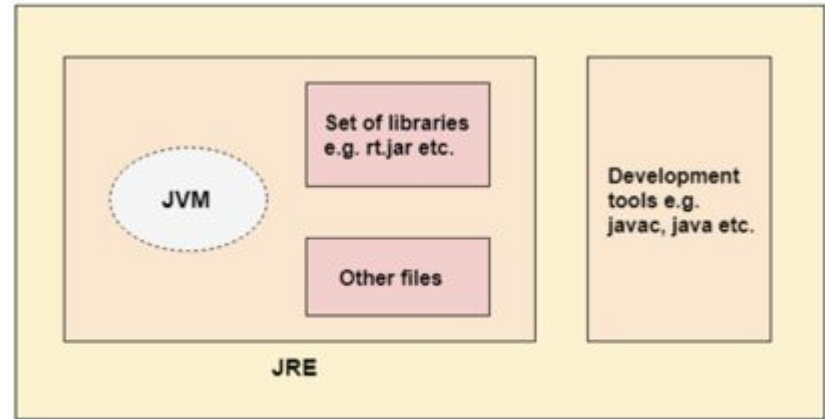
JavaFX

JDK, JRE, JVM Kavramları

JVM, Java Byte koda dönüştürülebilen her yazılım geliştirme dilini çalıştırabilme yeteneğine sahiptir.

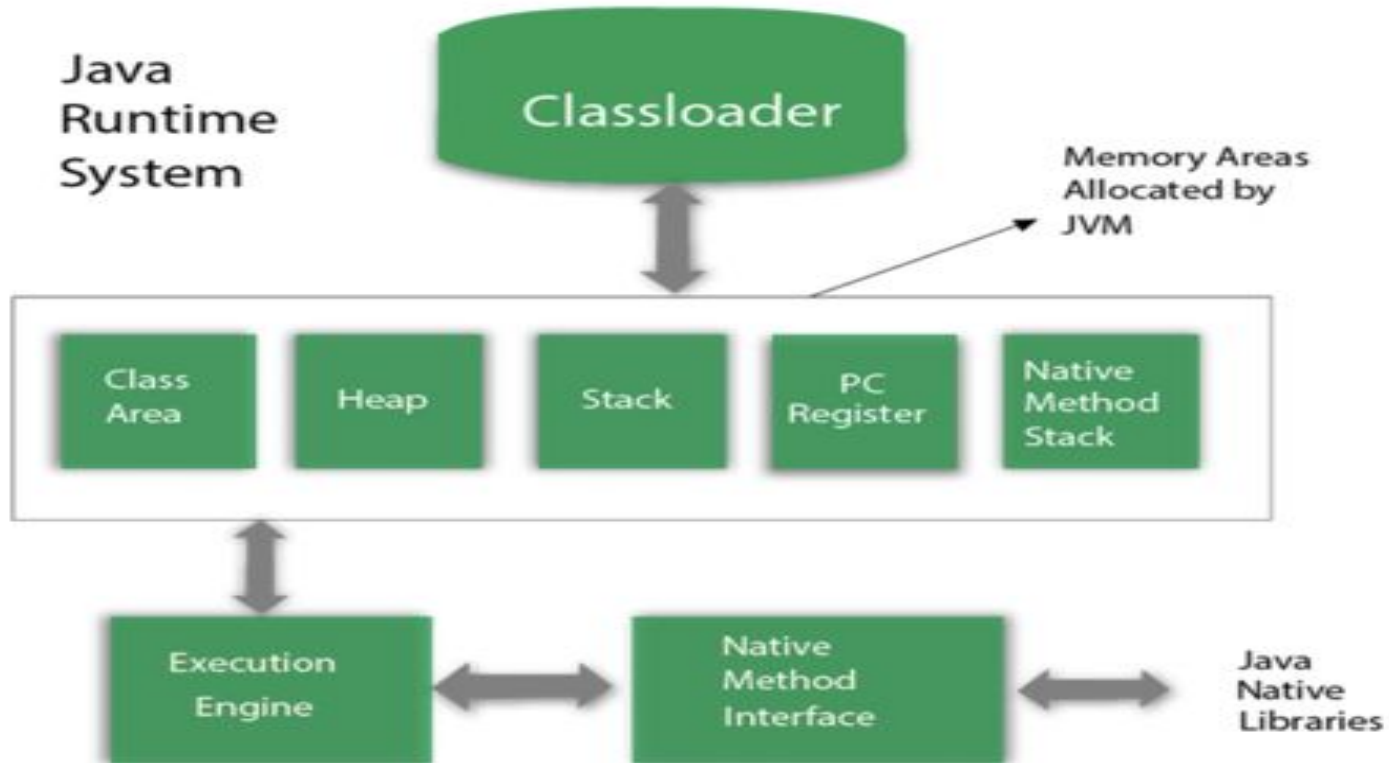


JRE

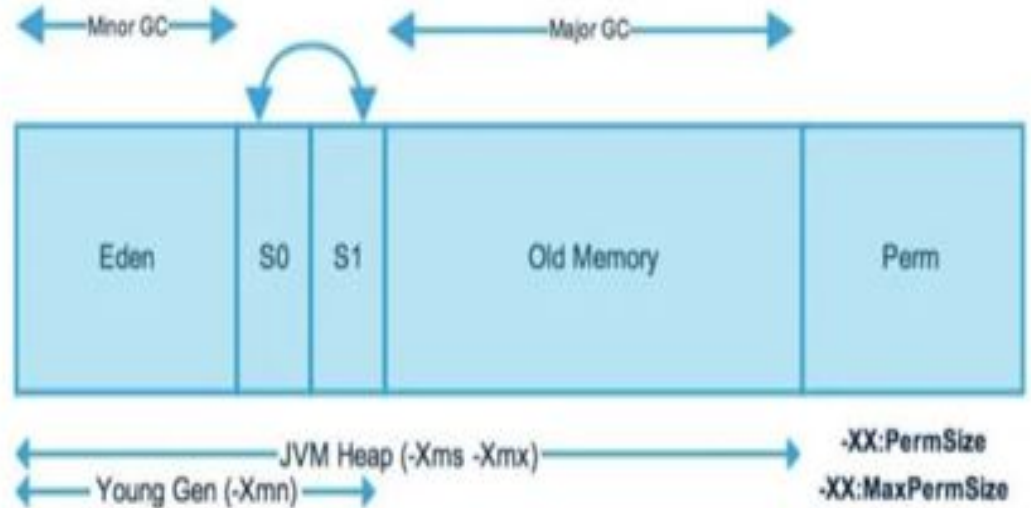
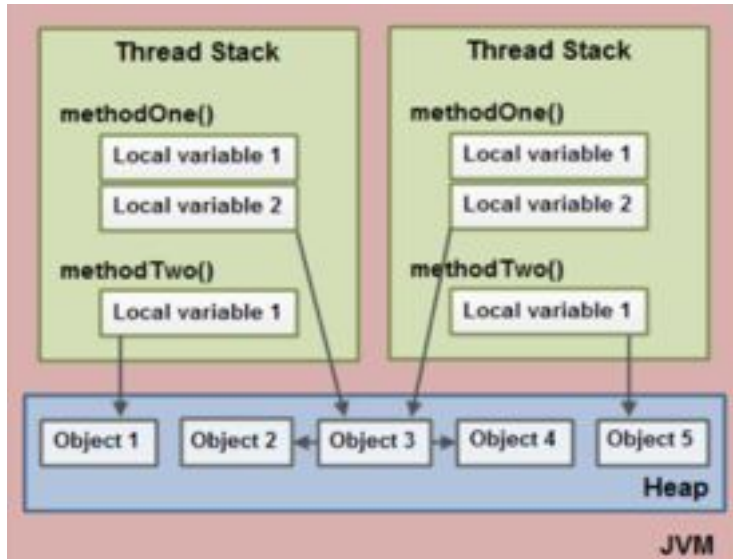


JDK

Java Virtual Machine Mimarisi



Java Hafıza Mimarisi



Java İlk Program

```
class MyFirstProgram {  
  
    public static void main(String args[]) {  
        System.out.println("Hello Java! | Merhaba Java!");  
    }  
  
}
```

Java Temel Kavramlar

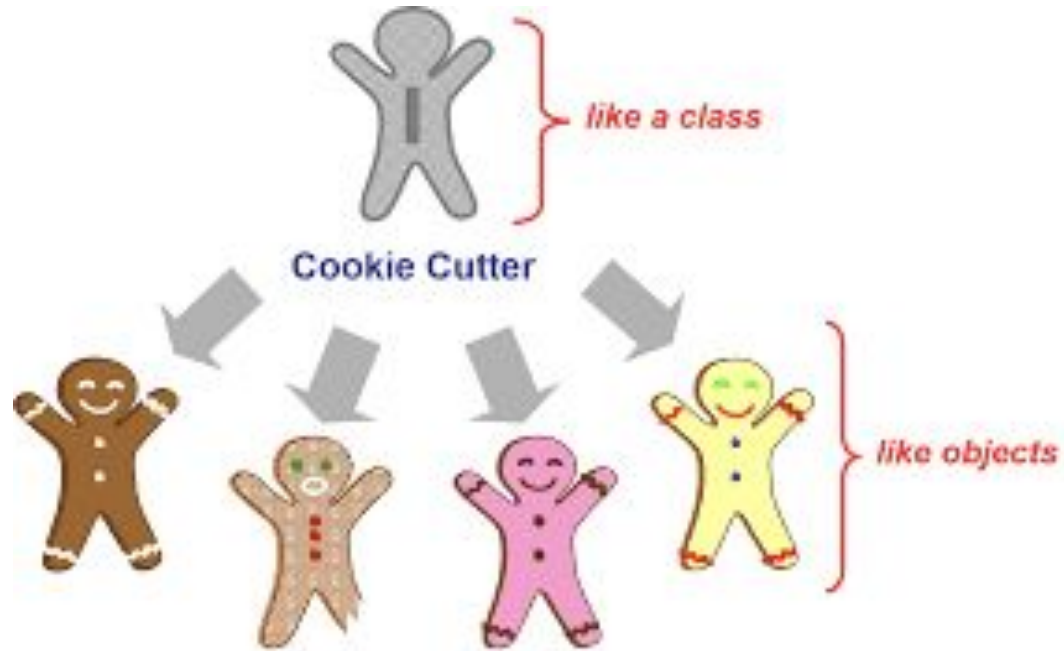
Sınıf: Gerçek hayatta yer alan bir nesnenin yazılım dünyasındaki model halidir.

Nesne: Tasarlanan bu modelden üretilenlere de nesne denilmektedir.

Metot (Fonksiyon): Fonksiyonlar eylemleri işaret eder. Örneğin: Printer isimli bir sınıfa ait “print” isimli eylem yazılım dünyasında bir metodu ifade eder. Metotlar belli parametreler alarak veya almadan belli bir üreten veya üretmeyen kod parçalarıdır.

Değişken: Değişkenler nesnenin niteliklerini işaret ederler. Örneğin: Printer isimli sınıfa ait “color” isimli nitelik bir değişkeni ifade eder. Eğer rengi beyazsa beyaz bir değer alır.

Java Temel Kavramlar



Java'da Değişkenler (Variables)

- Yerel Değişkenler
- Sınıf Değişkenleri (Statik olanlar)
- Nesne Değişkenleri (Statik olmayanlar)

Java Anahtar Sözcükler (Reserved Keywords)

abstract	assert	boolean	break
byte	case	catch	char
class	const	continue	default
do	double	else	enum
extends	final	finally	float
for	goto	if	implements
import	instanceof	int	interface
long	native	new	package
private	protected	public	return
short	static	strictfp	super
switch	synchronized	this	throw
throws	transient	try	void
volatile	while		

Java'da Sınıflar (Classes)

```
public class Dog {  
  
    String breed;  
    int age;  
    String color;  
  
    void barking() {  
    }  
  
    void hungry() {  
    }  
  
    void sleeping() {  
    }  
}
```

Java'da Nesneler (Objects)

Gerçek hayata döndüğümüzde etrafımızda yüzlerce nesne görürüz. Aslında, her nesnenin var olan bir durumu ve davranışı vardır.

Örneğin: bir köpeği ele aldığımızda rengi, ismi, cinsi köpeğe ait durumu ifade eder. Havlaması, koşmak, acıkması ise onun davranışlarını ifade eder.

Java'da Kod Kapsam Blokları

```
public class ConnectionPool
{ // Sınıf kapsamının başlangıcı

    int connectionMaximumLimit = 50; // Nesne değişkenidir.

    static int currentActiveConnectionCount = 10; //static değişkendir.
    Sınıf değişkenidir.

    public void acquireConnection()
    { // metot (fonksiyon) kapsamının başlangıcı

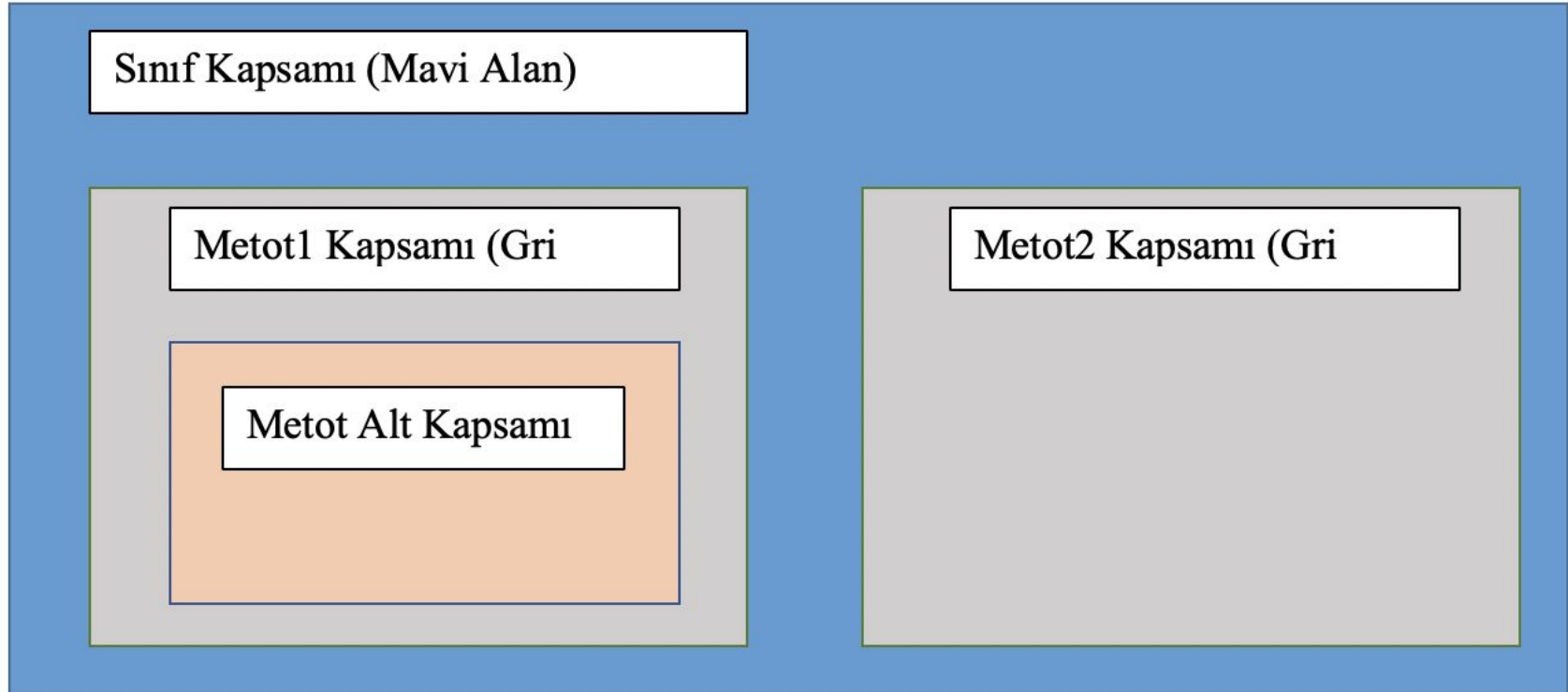
        int processId = 90; // Yerel değişkendir.

        // Diğer kodlar ...

    } // metot (fonksiyon) kapsamının sonu

} // Sınıf kapsamının sonu
```

Java'da Kod Kapsam Blokları



Java'da Sınıf Kurucuları (Constructors)

```
public class Puppy {  
  
    public Puppy() {  
        // parametresiz bir kurucu metot.  
    }  
  
    public Puppy(String name) {  
        // name isminde bir değişkenle tanımlanmış bir kurucu metot.  
    }  
}
```

Java'da Sınıf Kurucuları (Constructors)

```
Puppy p = new Puppy();
```

```
Puppy p = new Puppy("Pamuk");
```

Java'da Paketler (Packages)

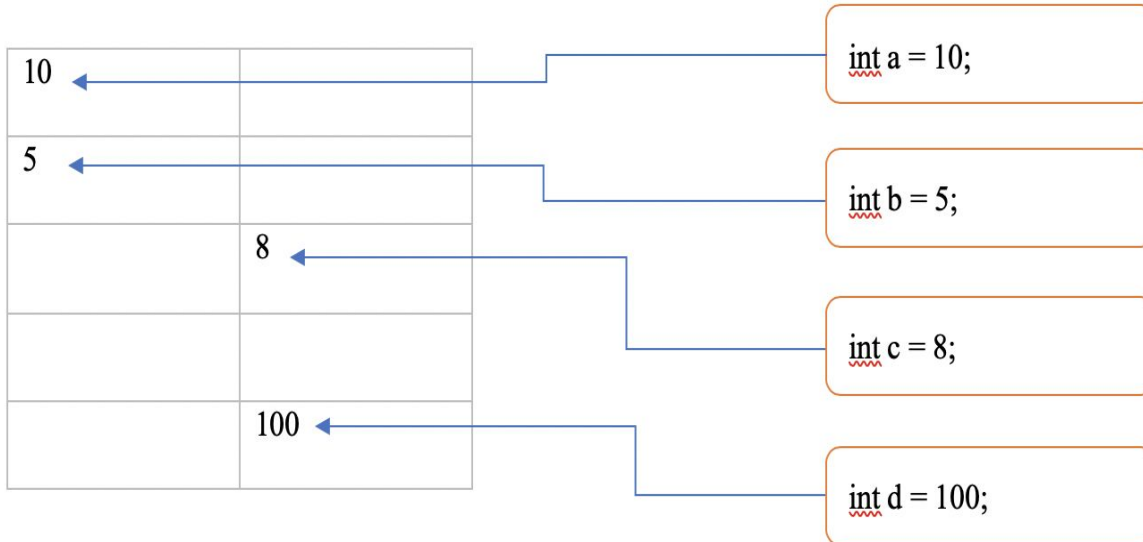
Java'da sınıfları (classes) ve arayüzleri (interfaces) bir araya toplamak için paket kavramı vardır.

Birbiriyle mantıksal ilişkiye sahip sınıflar bir paket altına hiyerarşik bir şekilde toplanır.

```
import java.io.*;
```

Java Değişkenler

`<veri tipi> <değişken ismi> = veri (değer)`

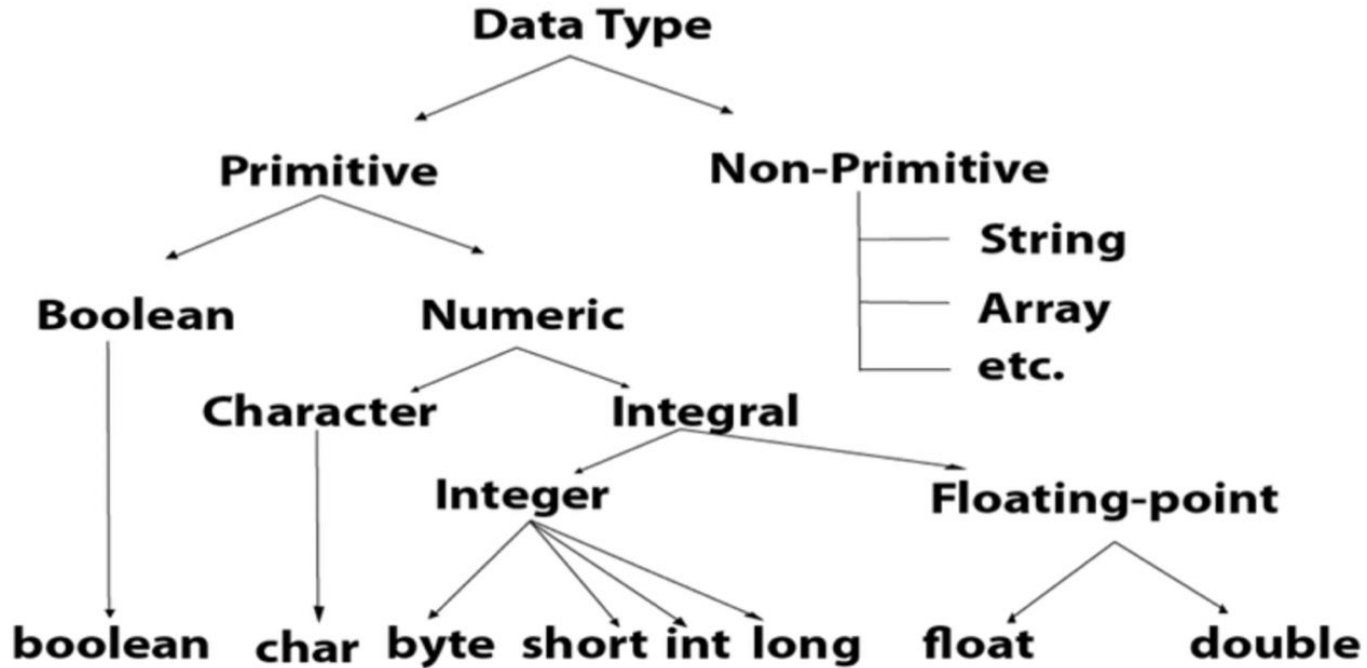


Java Değişkenler

`<veri tipi> <değişken ismi> = veri (değer)`

```
int a, b, c;           // 3 tane değişken virgüller ile ayrılarak tek satırda
                        tanımlanabilir.
int a = 10, b = 10;    // Birden fazla değişken aynı satırda ilk değerleri
                        atanarak tanımlanabilir.
byte b = 22;           // Tek değişkene ilk değer ataması yapılarak
double pi = 3.14159;   // Tek değişkene ilk değer ataması yapılarak
char a = 'a';          // Tek değişkene ilk değer ataması yapılarak
```

Java Veri Tipleri



Java Temel Operatörler

- Aritmetiksel Operatörler
- İlişkisel ve Eşitlik Operatörler
- Bitsel (Bit Düzeyinde) Operatörler
- Mantıksal Operatörler
- Atama Operatörleri

Java Temel Operatörler

```
public static void main(String args[]) {  
    int a = 10;  
    int b = 20;  
    int c = 25;  
    int d = 25;  
  
    System.out.println("a + b = " + (a + b) );  
    System.out.println("a - b = " + (a - b) );  
    System.out.println("a * b = " + (a * b) );  
    System.out.println("b / a = " + (b / a) );  
    System.out.println("b % a = " + (b % a) );  
    System.out.println("c % a = " + (c % a) );  
    System.out.println("a++    = " + (a++) );  
    System.out.println("b--    = " + (a--) );  
  
    System.out.println("d++    = " + (d++) );  
    System.out.println("++d    = " + (++d) );  
}
```

Aritmetik Operatörleri

```
public static void main(String args[]) {  
    int a = 10;  
    int b = 20;  
    int c = 25;  
    int d = 25;  
  
    System.out.println("a + b = " + (a + b) );  
    System.out.println("a - b = " + (a - b) );  
    System.out.println("a * b = " + (a * b) );  
    System.out.println("b / a = " + (b / a) );  
    System.out.println("b % a = " + (b % a) );  
    System.out.println("c % a = " + (c % a) );  
    System.out.println("a++    = " + (a++) );  
    System.out.println("b--    = " + (a--) );  
  
    System.out.println("d++    = " + (d++) );  
    System.out.println("++d    = " + (++d) );  
}
```

İlişkisel ve Eşitlik Operatörleri

```
public static void main(String args[]) {  
    int a = 10;  
    int b = 20;  
  
    System.out.println("a == b = " + (a == b) );  
    System.out.println("a != b = " + (a != b) );  
    System.out.println("a > b = " + (a > b) );  
    System.out.println("a < b = " + (a < b) );  
    System.out.println("b >= a = " + (b >= a) );  
    System.out.println("b <= a = " + (b <= a) );  
}
```

Bitsel Operatörler

```
public static void main(String args[]) {  
    int a = 60;          /* 60 = 0011 1100 */  
    int b = 13;          /* 13 = 0000 1101 */  
    int c = 0;  
  
    c = a & b;           /* 12 = 0000 1100 */  
    System.out.println("a & b = " + c );  
  
    c = a | b;           /* 61 = 0011 1101 */  
    System.out.println("a | b = " + c );  
  
    c = a ^ b;           /* 49 = 0011 0001 */  
}
```

Bitsel Operatörler

```
System.out.println("a ^ b = " + c );

c = ~a;                /*-61 = 1100 0011 */
System.out.println("~a = " + c );

c = a << 2;            /* 240 = 1111 0000 */
System.out.println("a << 2 = " + c );

c = a >> 2;            /* 15 = 1111 */
System.out.println("a >> 2 = " + c );

c = a >>> 2;           /* 15 = 0000 1111 */
System.out.println("a >>> 2 = " + c );
}
```


Mantıksal Operatörler

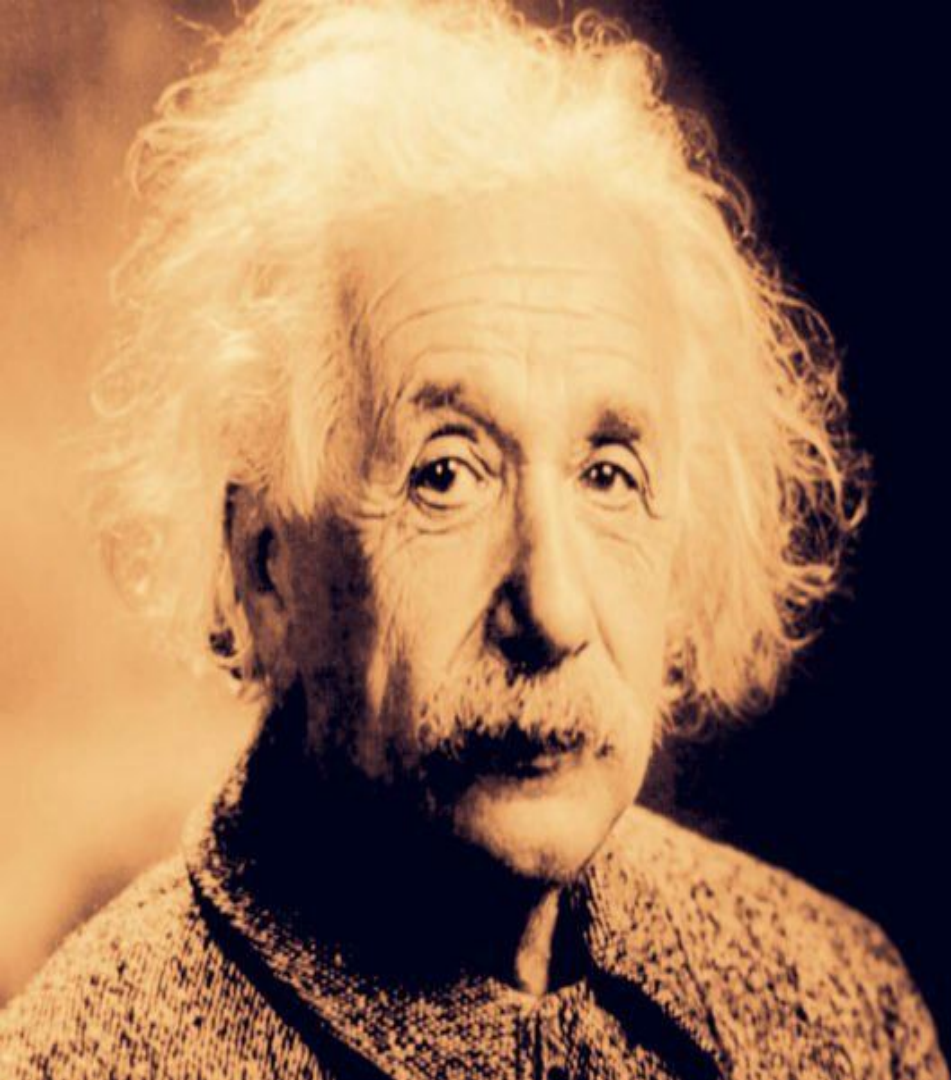
```
public class Test {  
  
    public static void main(String args[]) {  
        boolean a = true;  
        boolean b = false;  
  
        System.out.println("a && b = " + (a&&b));  
        System.out.println("a || b = " + (a||b));  
        System.out.println("!(a && b) = " + !(a && b)); } }
```

Soru İşareti Operatörü

```
public class Test {  
  
    public static void main(String args[]) {  
        int a, b;  
        a = 10;  
        b = (a == 1) ? 20: 30;  
        System.out.println( "Value of b is : " + b );  
  
        b = (a == 10) ? 20: 30;  
        System.out.println( "Value of b is : " + b );  
    }  
}
```

“instanceof” Operatörü

```
public class Test {  
  
    public static void main(String args[]) {  
  
        String name = "James";  
  
        boolean result = name instanceof String;  
        System.out.println( result );  
    }  
}
```



“We now accept the fact
that learning is a lifelong
process of keeping
abreast of change.”

Albert Einstein

Teşekkürler!



/in/batuhanduzgun



@batux

batuhan.duzgun@windowslive.com

www.batuhanduzgun.com

