

# Free Java Dersleri

## Ders - 01

### Programlamaya Giriş

## Programlama Dili Nedir?

Programlama dili, yazılımcının bir **algoritmayı** ifade etmek amacıyla, bir **bilgisayara** ne yapmasını istediğini anlatmasının **tektipleştirilmiş** yoludur.



Programlama dilleri, yazılımcının bilgisayara hangi veri üzerinde işlem yapacağını, verinin nasıl depolanıp iletileceğini, hangi koşullarda hangi işlemlerin yapılacağını tam olarak anlatmasını sağlar.

Programlama dilleri sayesinde bir bilgisayarın hangi durumda ne çeşit çıktı verebileceği kontrol edilebilir.

Kısacası programlama dilleri sayesinde bilgisayarlar ve insanlar verimli bir iletişim sağlayabilirler.

# Programlama Dili Nedir?

Bilgisayar'ın bizim istedigimiz seyi yapabilmesi icin



1- Bilgisayar'ın anlayacagi dili bizim bilmemiz



2- Yazdigimiz kodlari bilgisayar tarafından anlasildigini bilmemiz



3- Bilgisayarın bizim için ürettiği sonuçları anlamlandırabilmemiz gerekir.

# Programlama Dili Nedir?

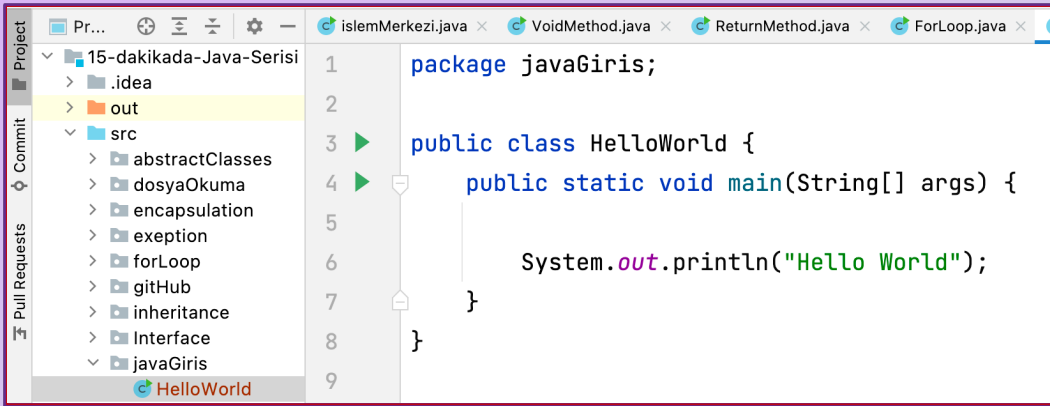
Kisaca ozetlersek, programlama dili bizimle bilgisayar arasindaki iletisimi saglayan dildir.

Peki..Bizimle bilgisayar arasindaki islemlerde patron kim ?

Kimin dedigi olur ?

Tabii ki bilgisayarlar bizim dedigimizi yaparlar

Ama bu istegimizi bilgisayarın anlayacağı özelliklerde yazmamız şartıyla.



```
1 package javaGiris;
2
3 public class HelloWorld {
4     public static void main(String[] args) {
5
6         System.out.println("Hello World");
7     }
8 }
9
```

Ornegin kodlarimizla, Hello World yazdirmak istiyorsak

```
110101010101010010100010101010
100101111000001000100101101010
101010010111000110
010001010010101110001000000101
0010101110001001010111
```

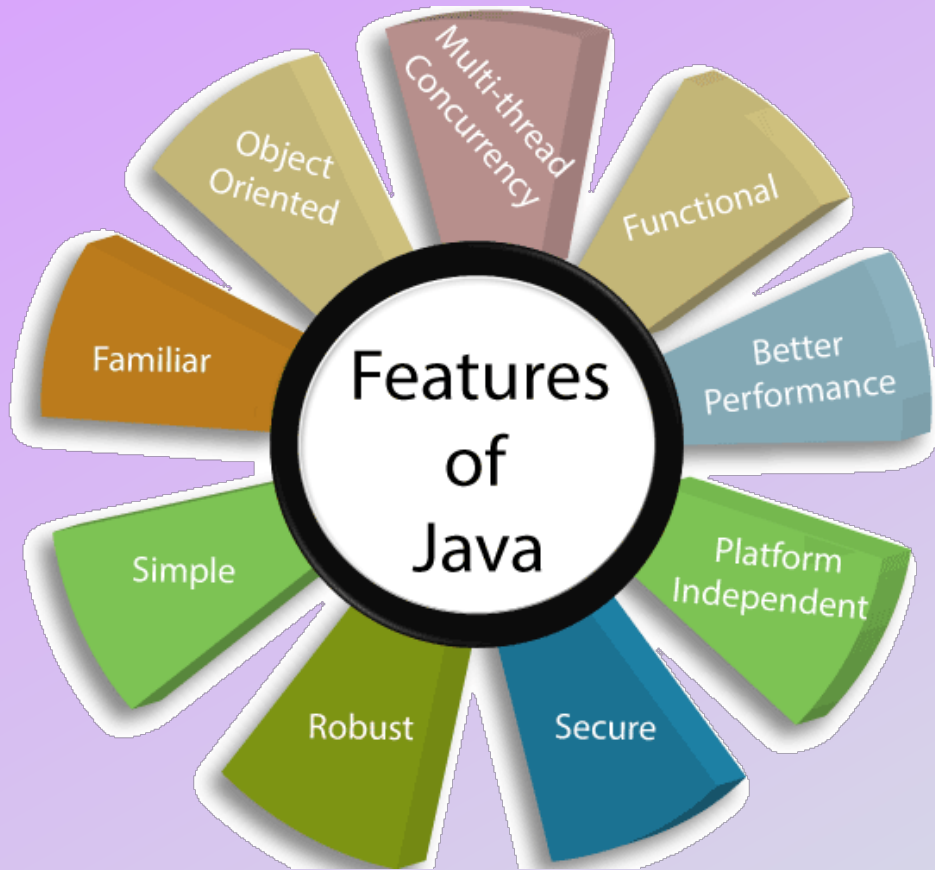
JDK Kodlari derler (Compile)

Binary kodlar

Islem



# Nicin Java?



Java, 1995 Yilinda ortaya cikan high-level, Object Oriented bir program olarak ortaya cikmistir.

Java'yi ilk gunden itibaren populer programlama dilleri arasinda one cikaran bazi ustunlukleri,

1- Ogrenmesi kolay

2- Dunyada en cok kullanan programlama dili

3 milyar mobil cihazda Java kullanıyor.

USA'de sirket bilgisayarlarinin %97'sinde, kişisel bilgisayarlarin %89'unda Java kullaniliyor

Linkedin, Uber, Netflix gibi pek cok populer uygulama Java tabanlı çalışıyor.

3- Güvenlidir

4- Platform independent'dir

5- Ücretsizdir.

Neden Java

<https://youtu.be/RUYASEnT6pU>



# Nicin Java ? / OOP Concept

6- Java, Object Oriented Programming konseptinde calisir.

OOP Konsept, kompleks programlari yapmaya kucuk parcalardan baslayip, sonra bunlari birlestirerek istenen sonuca ulasmaya denir.



Java'da, once Object (Nesne) olusturmaliyiz.

Her obje'nin 2 tur ozelligi olur.

- Feature (variable) – renk, pin sayisi vb...
- Functionality (method) – bizim girdigimiz degere gore degeri degisen ozellikler. Kanatli, tekerli vb..

# Object Nasıl Olusturulur ?

Objeleri olusturabilmek icin oncelikle kalip olusturmaliyiz.



Kalip varsa, bu kaliptan istedigimiz kadar obje uretebilir ve bu objeleri birlestirip istedigimiz uygulamayi elde edebiliriz.

Java'da obje olusturabilmemiz icin kullanmamiz gereken kalip Class( Sinif )'dir.

Her bir obje bir Class'tan tureilmistir ve Class olmadan obje uretmek mumkun degildir.

# Class Hangi Bolumlerden Olusur?

```
public class C2_MethodCreation2 {
```

```
} // Class sonu
```

Bir Class'da temel 3 bolum bulunur.

1- Class Declaration

2- Curly Braces : Suslu Parantez

3- Class Body : Suslu parantezler arasinda kalan, kodlarimizi yazdigimiz bolum.



# Class Body'sinde Neler Bulunur ?

```
public class HelloWorld {  
  
    int ogrNo=1013;  
    String isim="Ali Can";  
    boolean ogrenciMi=true;  
    double notOrt=87.5;  
  
    public static void main(String[] args) {  
  
        double yaziliNotu=89;  
        double sozluNotu=92;  
  
    }  
  
    public void baskaMethod(){  
  
    }  
}
```

Bir Class Body'sinde variable ve method'lar bulunur.

1- Main Method

2- Variables

3- method'lar

# Main Method Nedir ?

```
public static void main(String[] args) {  
  
}
```

Main Method, Java'nin çalışmaya başladığı giriş noktasıdır (Entry point)

Main method oluşturulurken kullanılacak syntax (yazılacak keyword veya metin) sabittir, değiştirilemez.

Parantez içine yazılan (**String[] args**) main method'un çalışması için gerekli argümanların olduğu bir array'dir ve mutlaka yazılmalıdır.

**NOT :** main method olmayan class'lar run edilemez (direk çalıştırılmaz).

```
3  ▶ public class HelloWorld {  
4  
5  ▶ public static void main(String[] k) {  
6  
7  
8  
9  }  
10 }  
11
```

```
3  public class HelloWorld {  
4  
5  
6  public void baskaMethod(){  
7  
8  
9  }  
10  
11 }
```

# IntelliJ'de Proje Olusturma

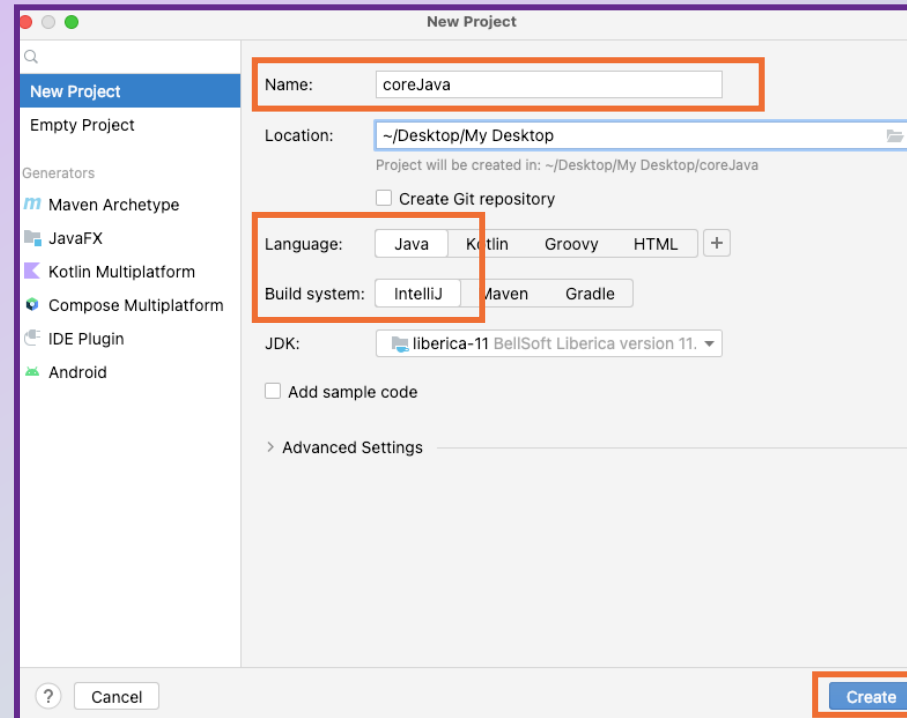
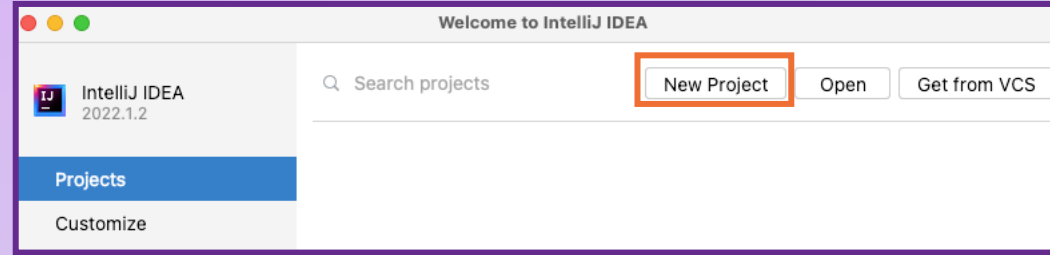
## IntelliJ Nedir?

Java gibi compiler programlar calismak icin ide( **I**ntegrated **D**evelopment **E**nvironment)'ye ihtiyac duyarlar.

Bircok ide olmakla birlikte, piyasada cok kullanildigi ve kod yazimini kolaylastirdigi icin biz intelliJ kullanacagiz.

## 1- Proje olusturma

IntelliJ'de **Projects** menusunda **New Project**'i secin,



Acilan menu'de

**Name** kismina projenin ismini yazin,

**Language** kisminde Java, **Build system** kisminde IntelliJ secili oldugunu kontrol edip,

**Create** butonuna basin.

# IntelliJ'de Proje Olusturma

## 2- Package(Class Dosyasi) olusturma

Acilan projede **src**'ye sag click yapip **New--Package**'i secip istedigimiz ismi yazin ve **finish**'e basin

## 3- Class olusturma

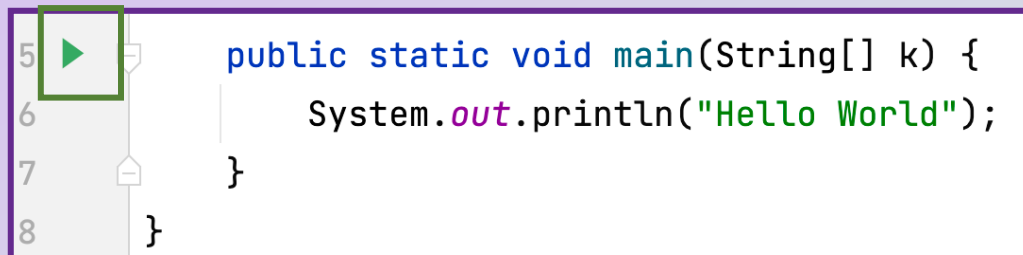
Acilan package ismine sag click yapip **New--Class**'i secip istedigimiz ismi yazin ve **finish**'e basin

## 4- Main method olusturma

Acilan Class icinde **main veya psvm** yazdigimizda, asagida cikan **main** yazisini sectiginizde IntelliJ bizim icin main method olusturacaktır.

## 5- Ilk kodumuzu yazdirma

Main method icerisinde **out** yazip acilan menuden **system.out.println**'i secin.



```
5  ▶ public static void main(String[] k) {  
6      System.out.println("Hello World");  
7  }  
8  }
```

Parantez icine **"Hello World"** yazin

**Yesil run tusu**'na basip class'i calistirin



## Class'a Kod Olmayan Yazı (Comment) Ekleme

Kod yazarken ilk hedef çalışan bir kod yazmaktır.

Ama asıl hedef çalışan ve Anlaşılabilir kod yazmaktır.

Kodlarımızı hem kendimiz hem de bizden sonra kodları kullanacak kişilerin daha iyi anlayabilmesi için, class'a açıklama cümleleri ekleyebiliriz .

```
5 ▶ public static void main(String[] k) {  
6  
7 // Tek satiri comment yapmak icin
```

En iyi Kod  
Çalışan Kod,  
Daha da iyisi  
Çalışan ve Anlaşılır Kod

```
10 /*  
11 Birden fazla  
12 satiri  
13 comment yapmak için  
14 */
```