

BBS515 Nesneye Yönelik Programlama

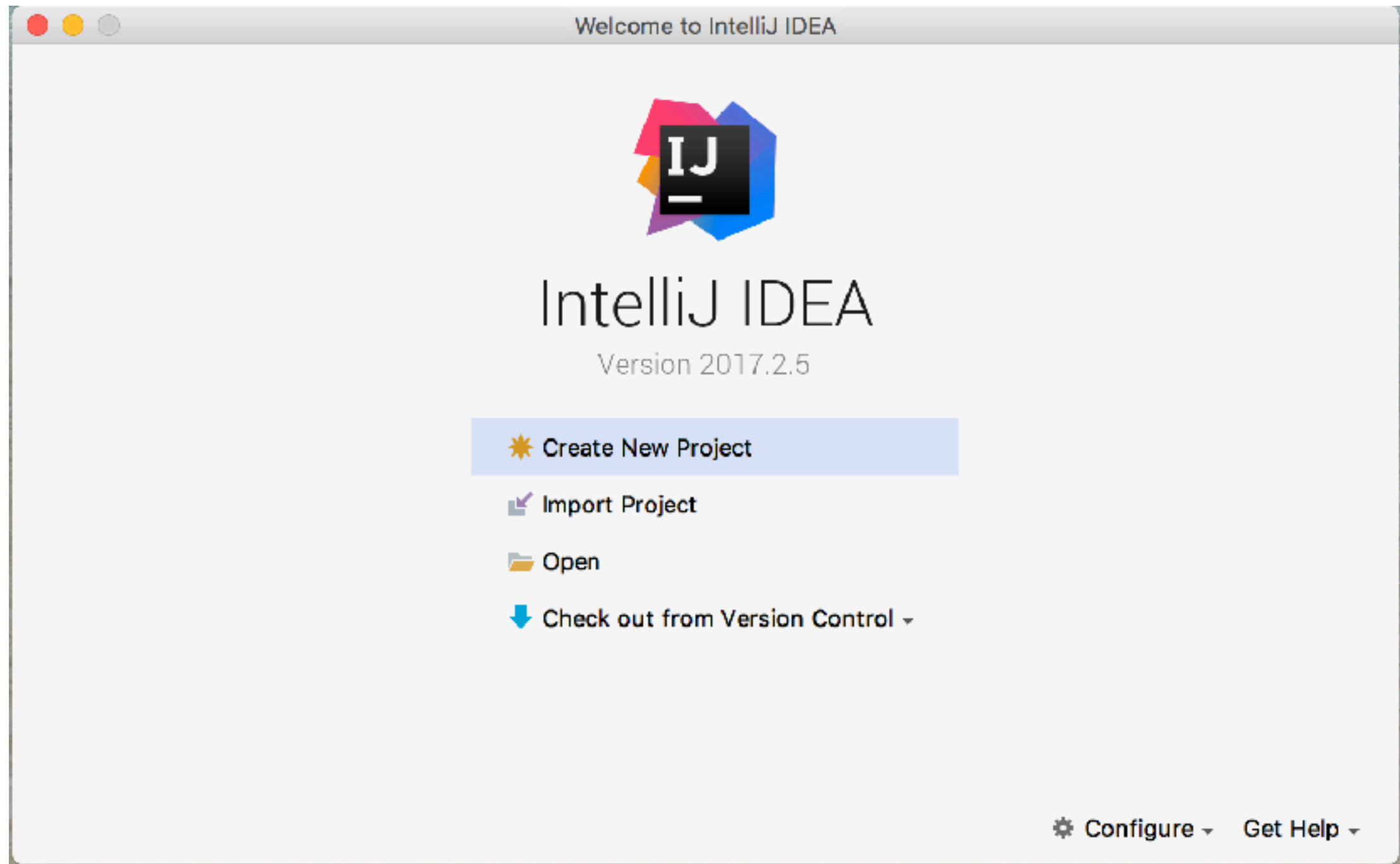
Ders 2

Zümra Kavafoğlu
<https://zumrakavafoglu.github.io/>

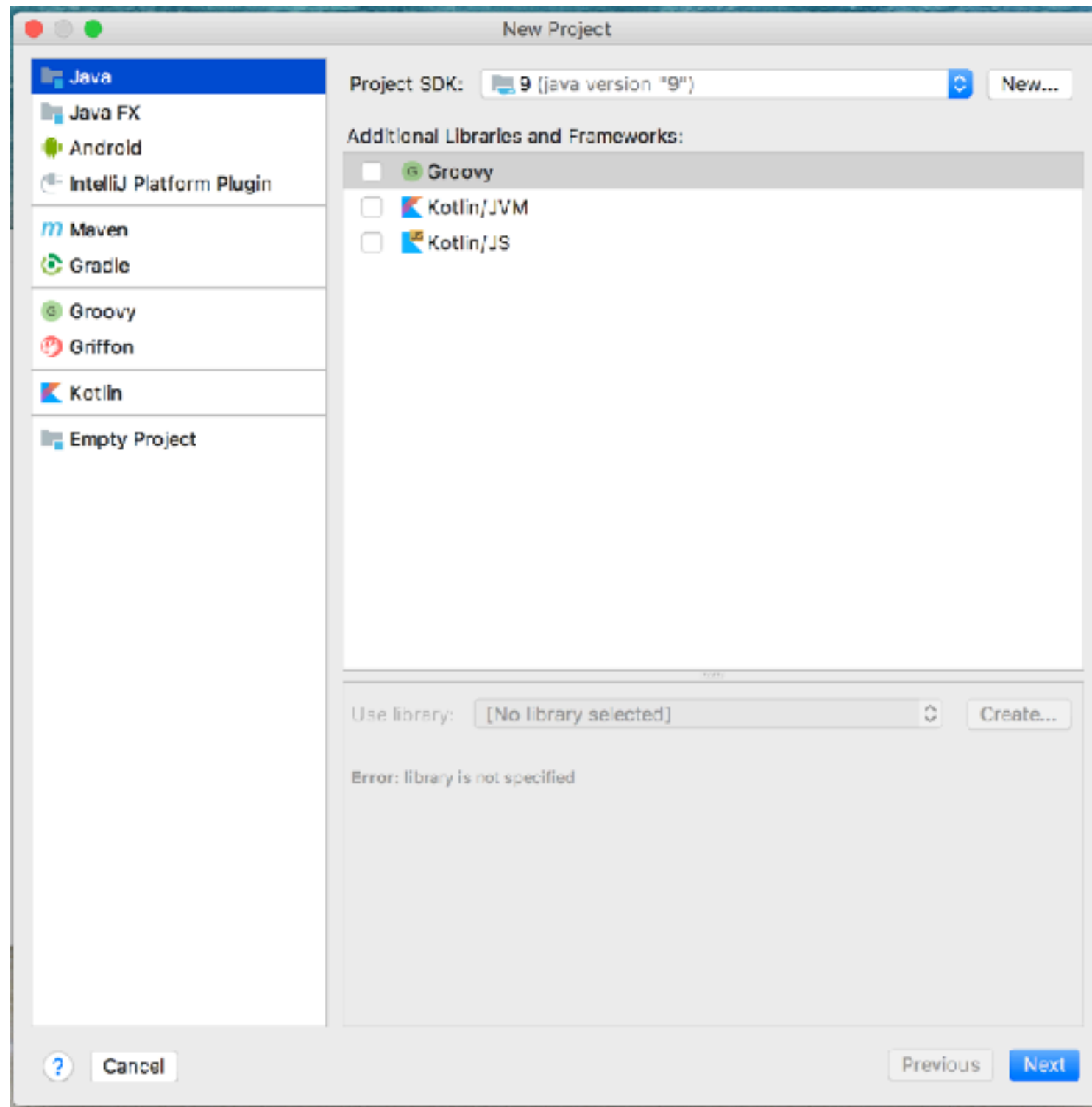
Bu derste neler öğreneceğiz?

- IntelliJ ile proje oluşturma
- Yorum satırları
- Sınıf deklarasyonu
- İsimlendirme kuralları
- Anahtar sözcükler
- main metodu
- İfadeler
- Derleme Hatası
- print ve println metodları
- Özel karakterler
- Değişken tanımlama ve değer atama
- Primitif veri tipleri
 - Nümerik veri tipleri
 - Boolean
 - char
- String veri tipi
- İkili nümerik operatörler
- Kısayol atama operatörleri
- Arttırma ve azaltma operatörleri
- Math sınıfı metodları
- printf
- Hafızada değişkenler
- final niteleyicisi
- Scanner sınıfıyla kullanıcı girdisi okuma
- Farklı nümerik veri tipleri arasında işlemler
- Veri tipi çevirme

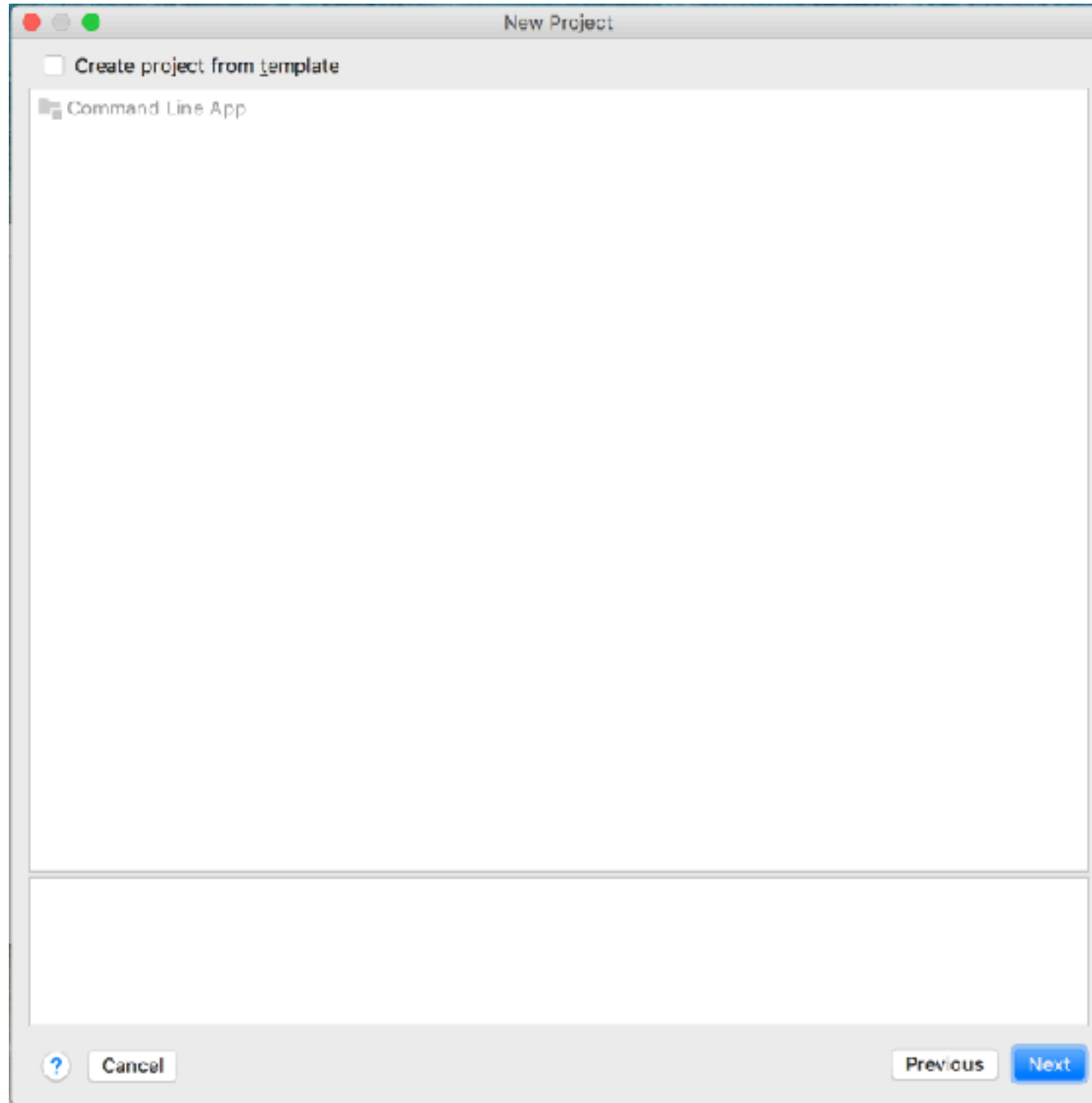
Java ile ilk program : IntelliJ ile proje oluřturmak



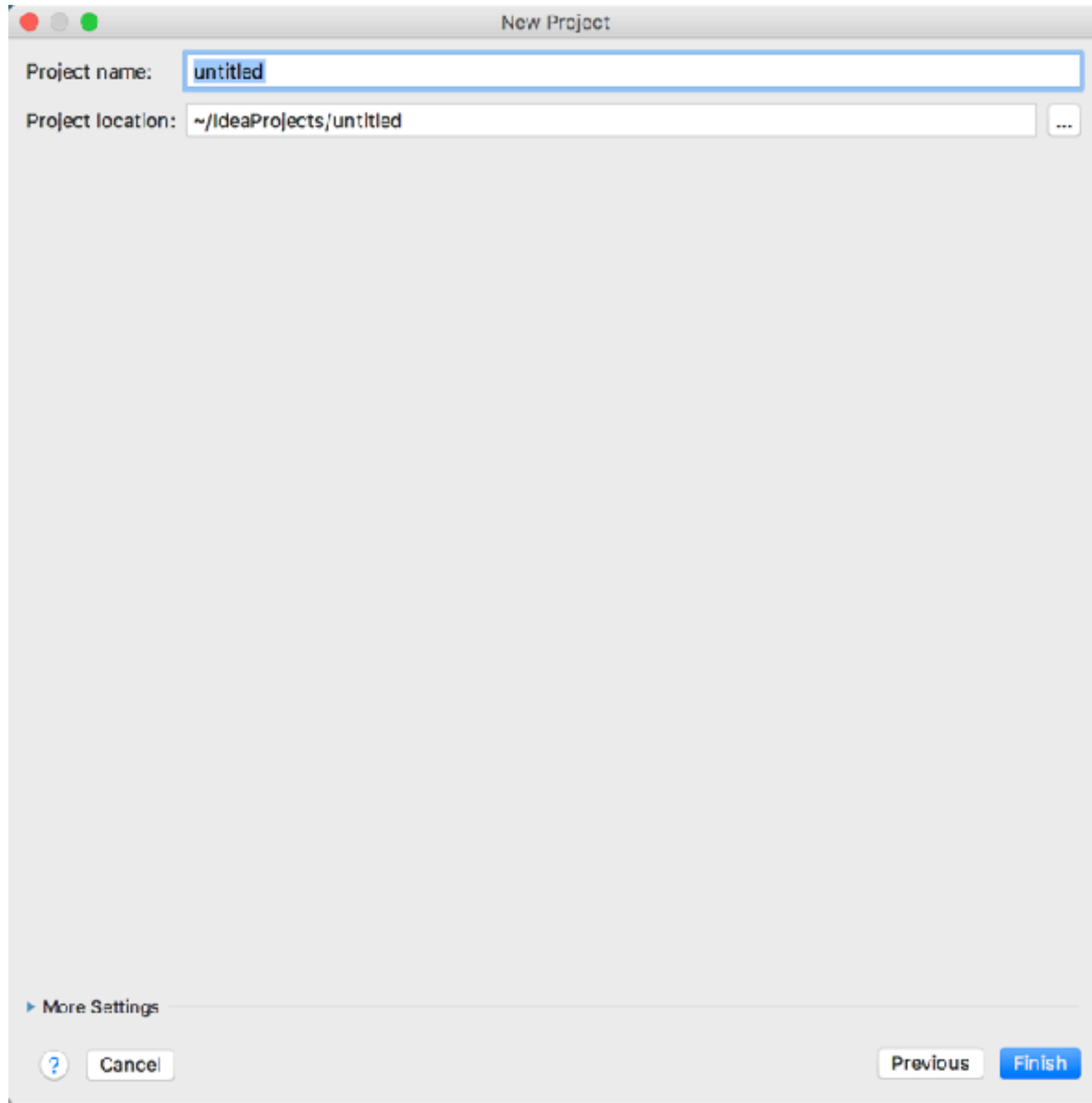
Java ile ilk program : IntelliJ ile proje oluřturmak



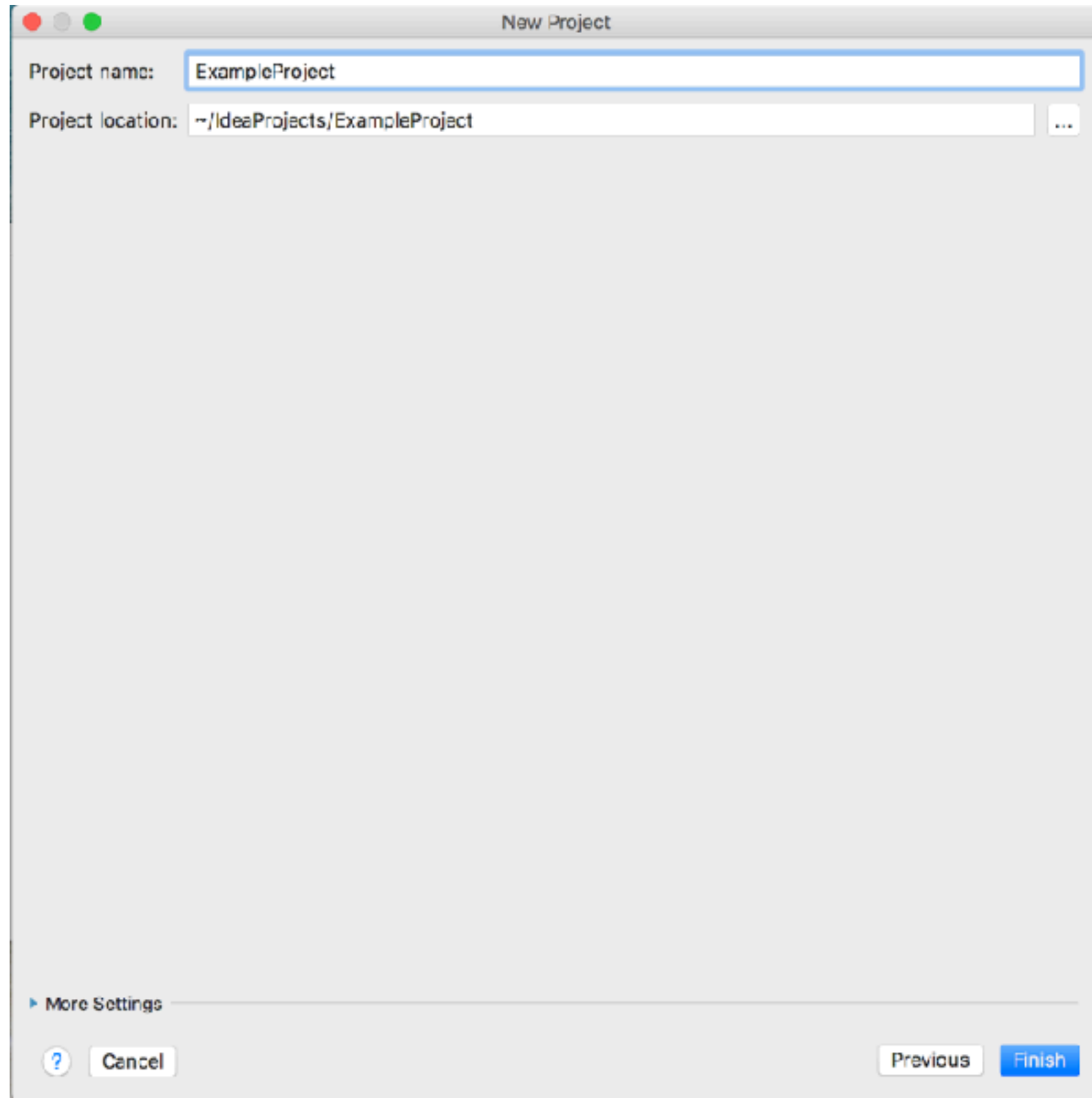
Java ile ilk program : IntelliJ ile proje oluřturmak



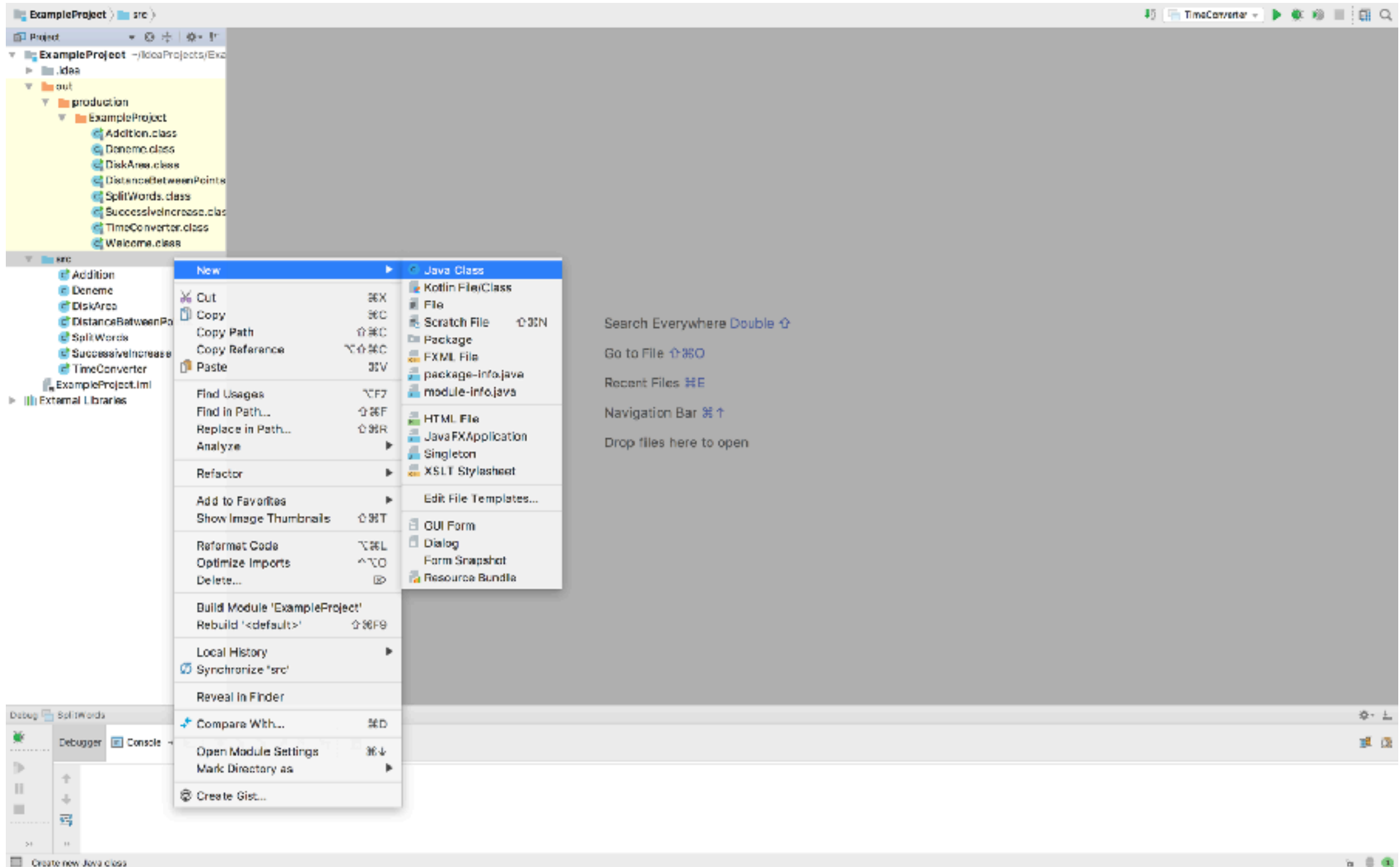
Java ile ilk program : IntelliJ ile proje oluşturmak



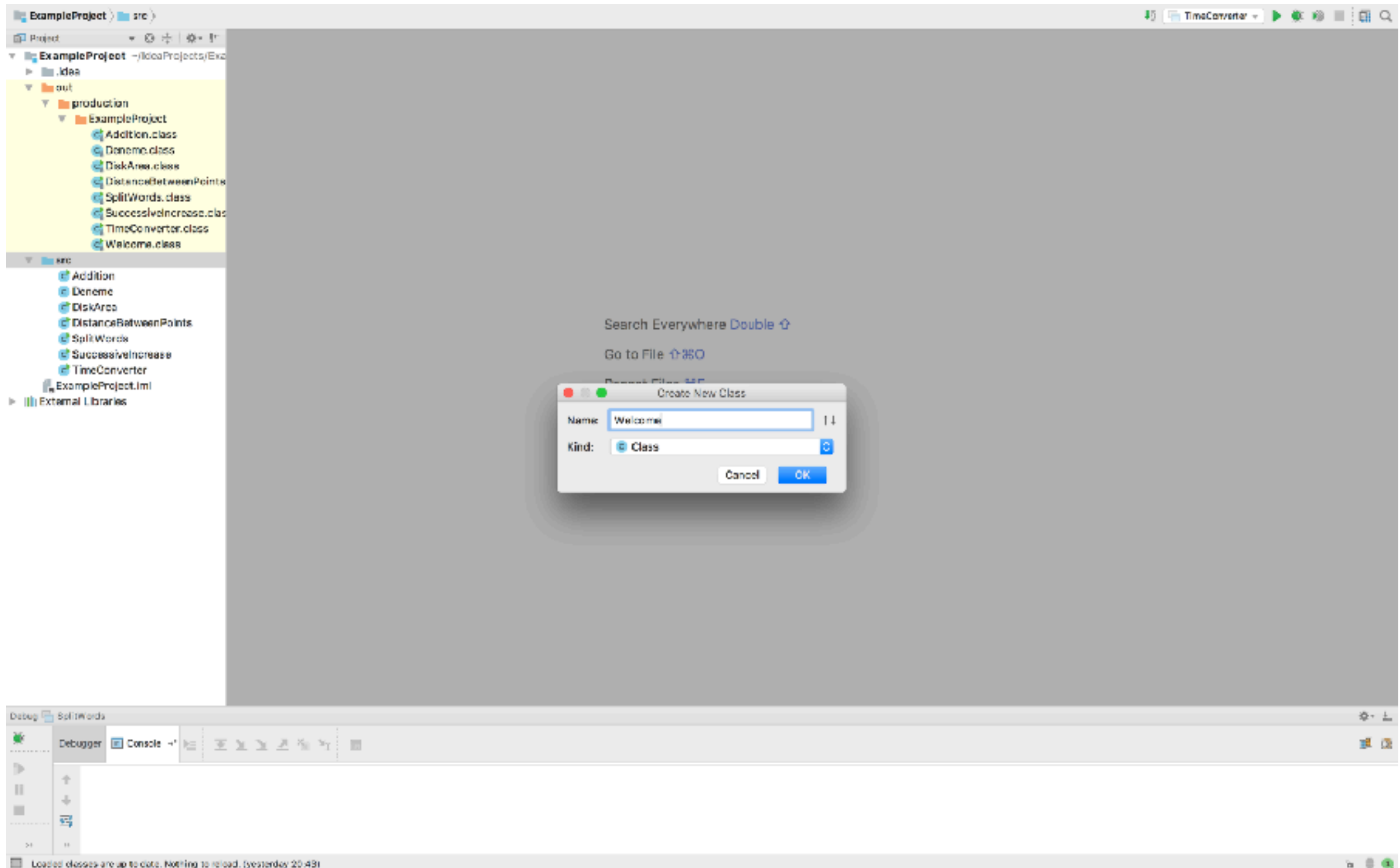
Java ile ilk program : IntelliJ ile proje oluřturmak



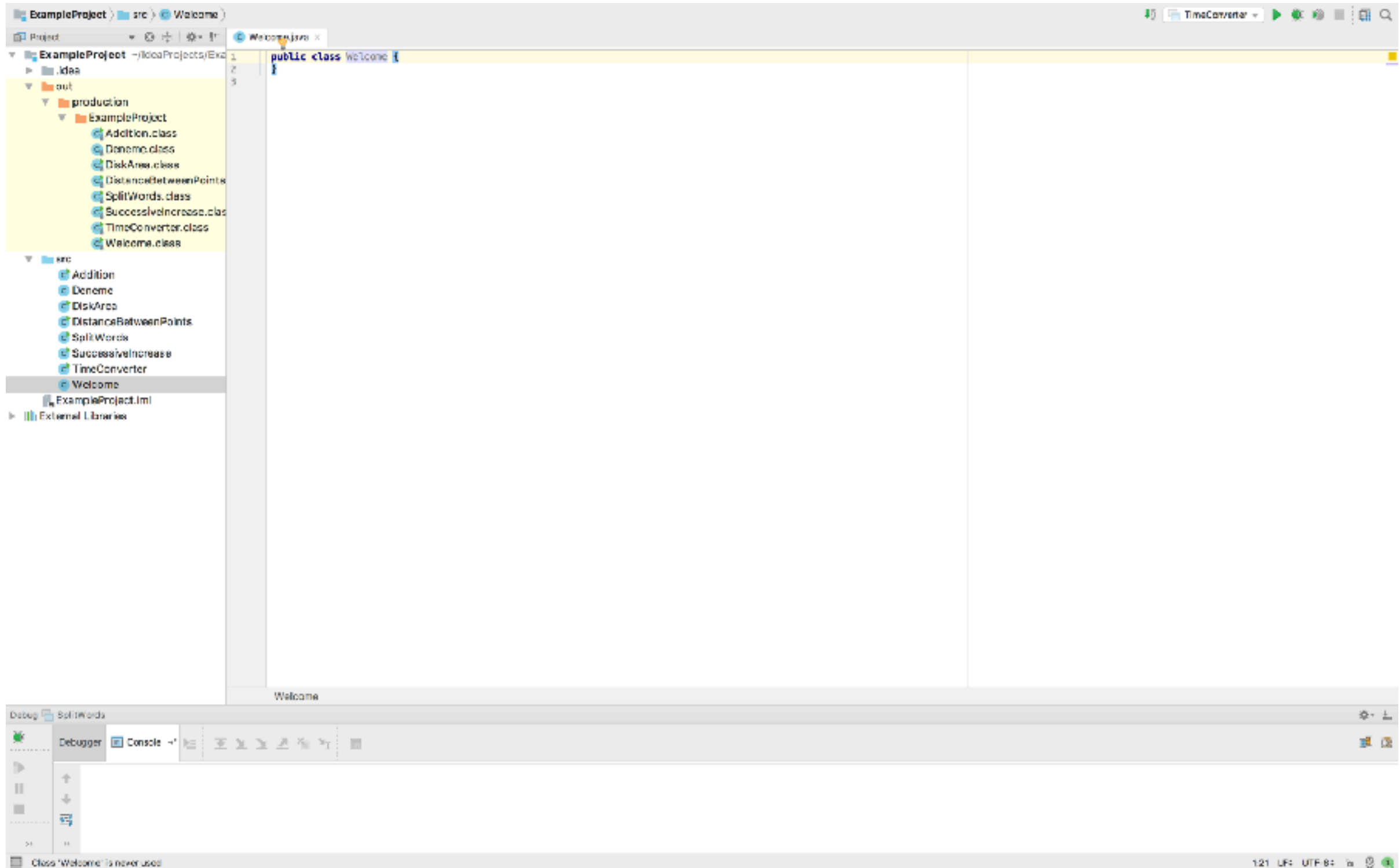
Java ile ilk program : Bir java dosyası oluşturmak



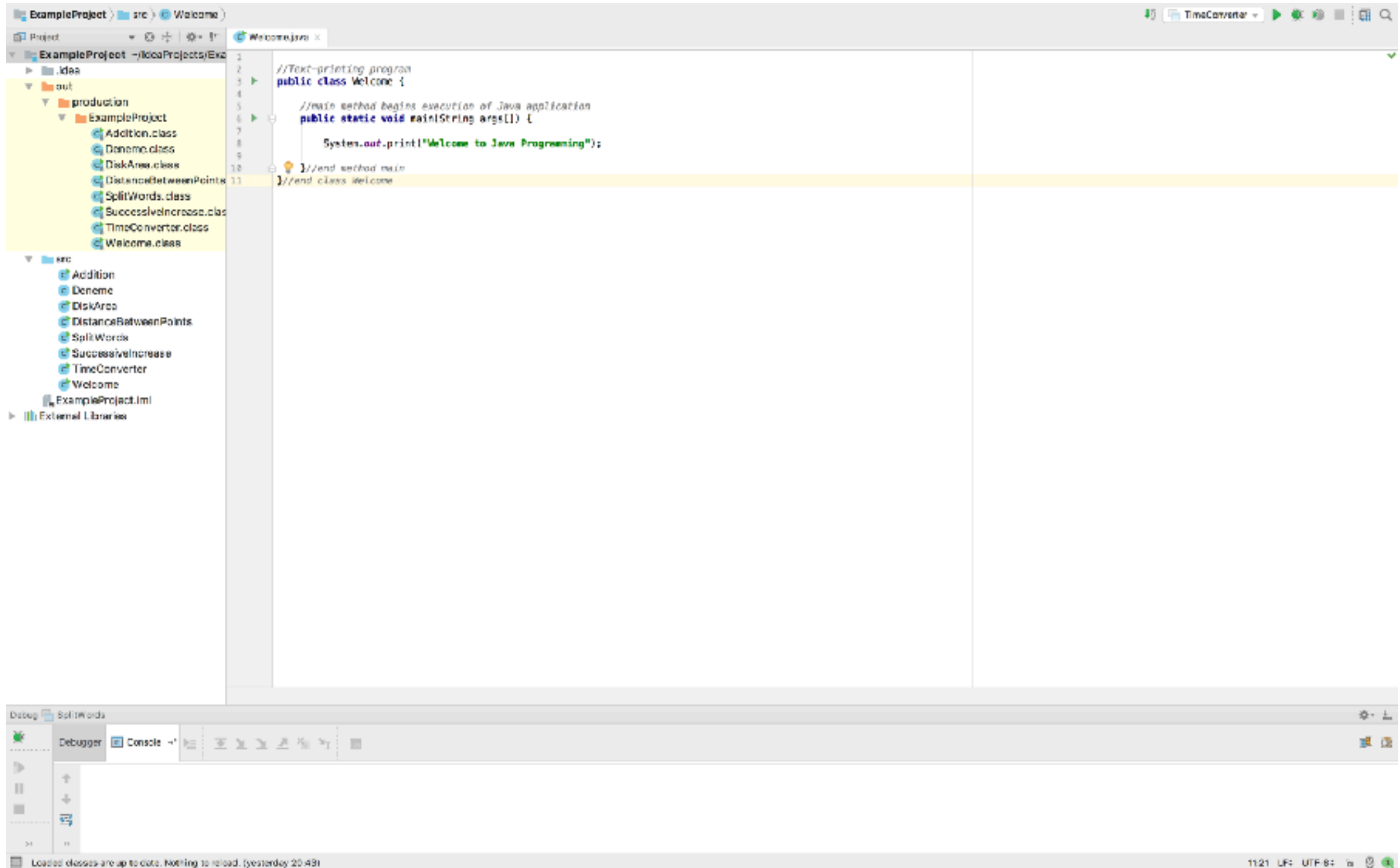
Java ile ilk program : Bir java dosyası oluşturmak



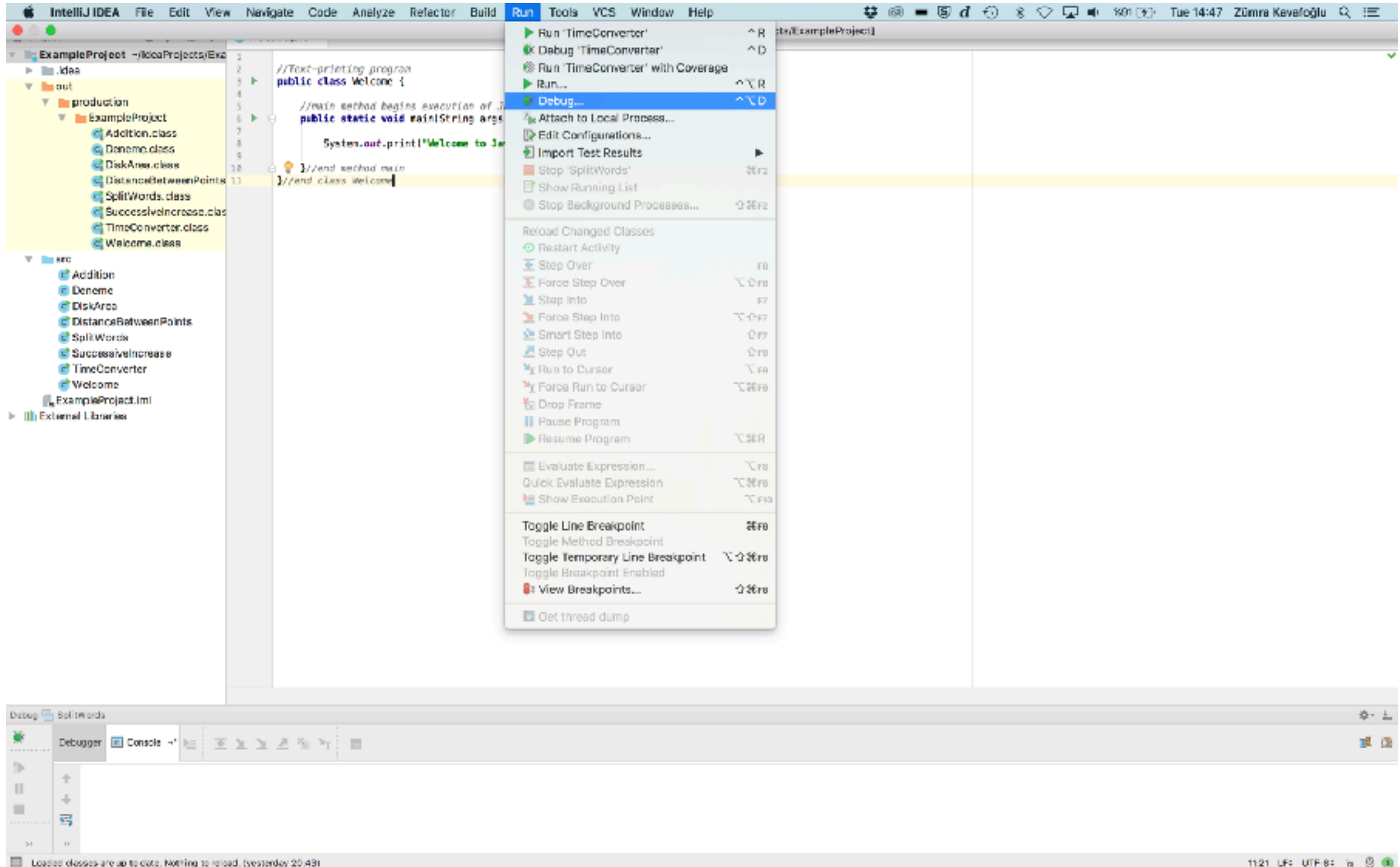
Java ile ilk program : Bir java dosyası oluşturmak



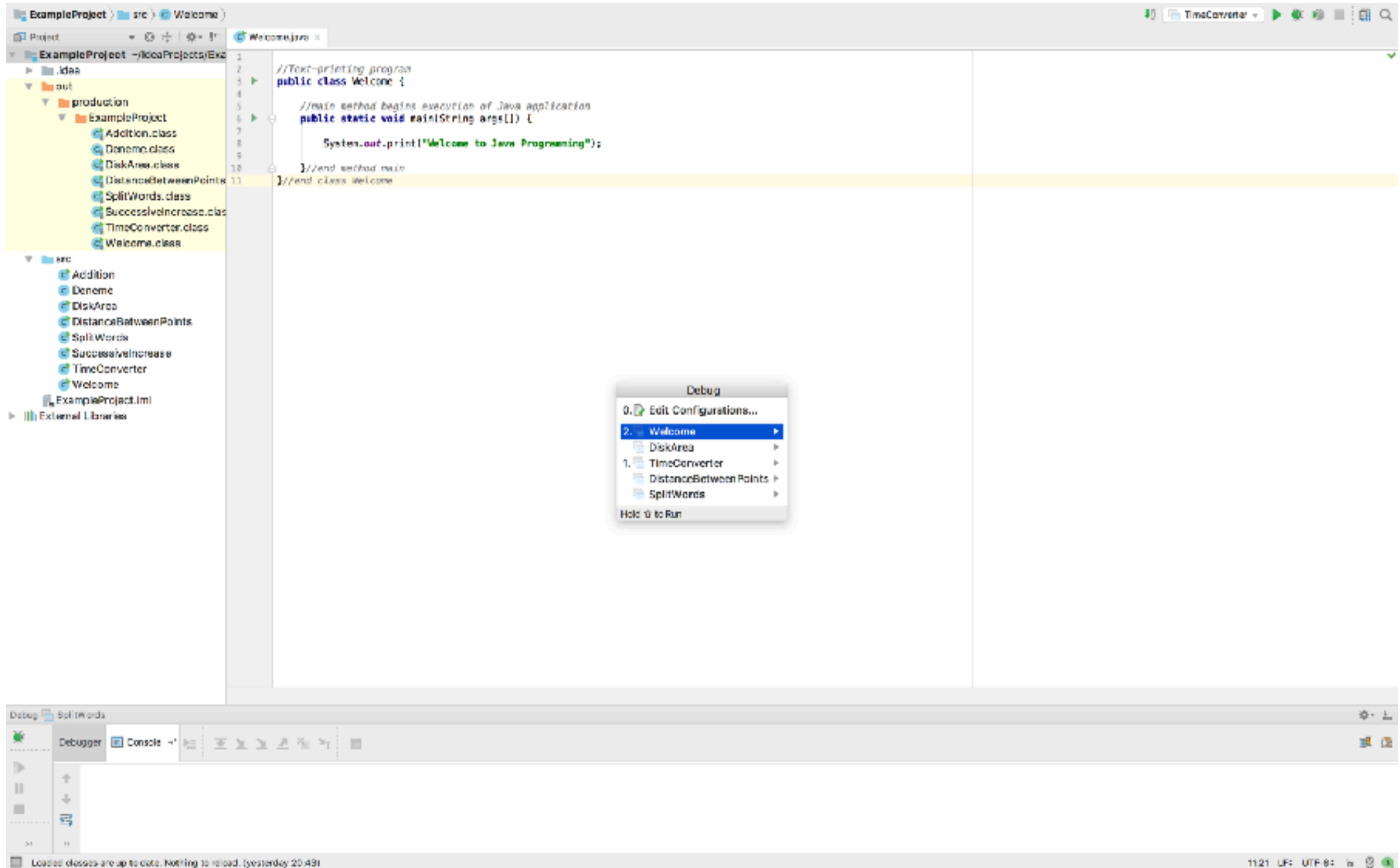
Java ile ilk program : Bir java dosyası oluşturmak



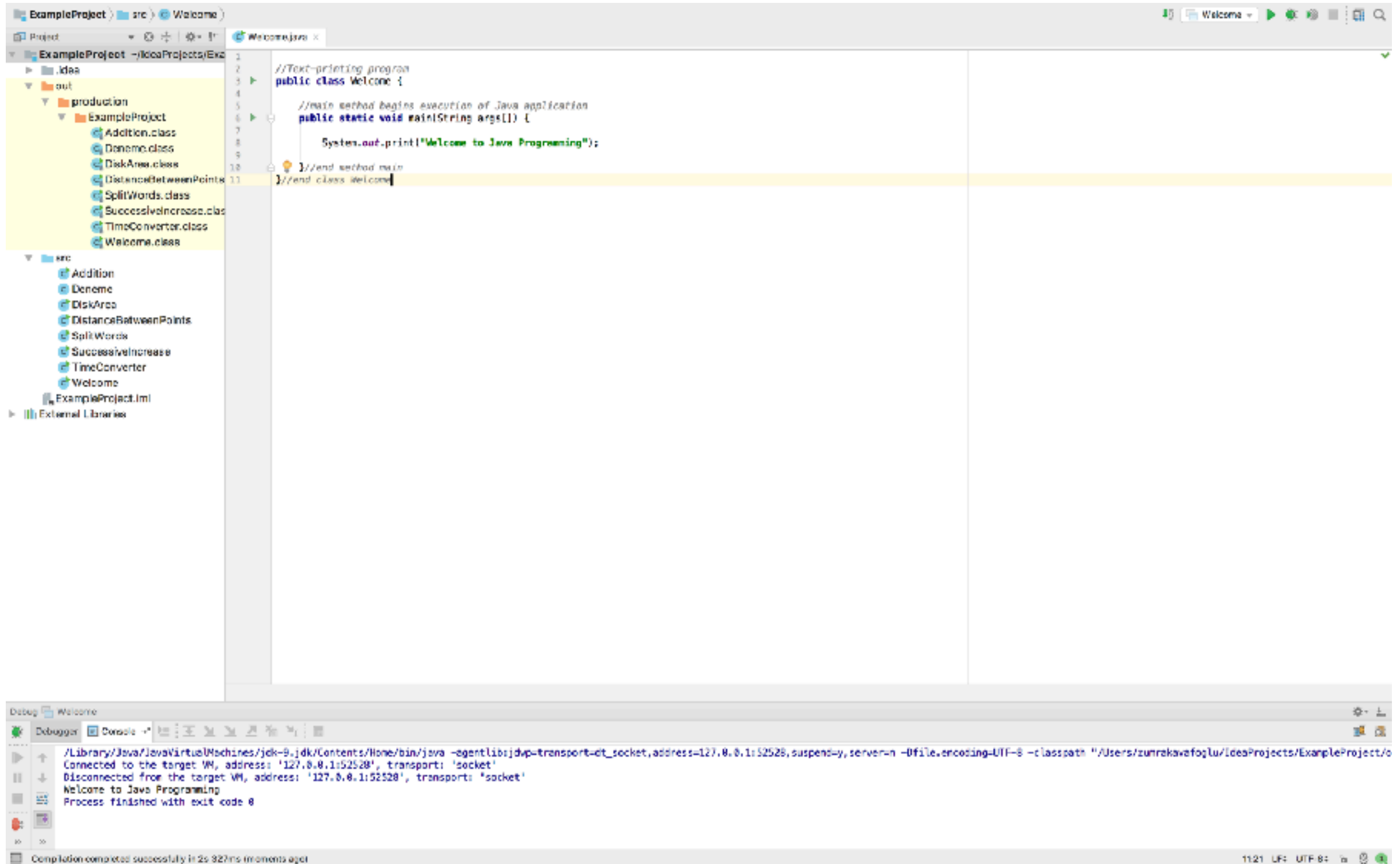
Java ile ilk program : Debug ile programı çalıştırmak



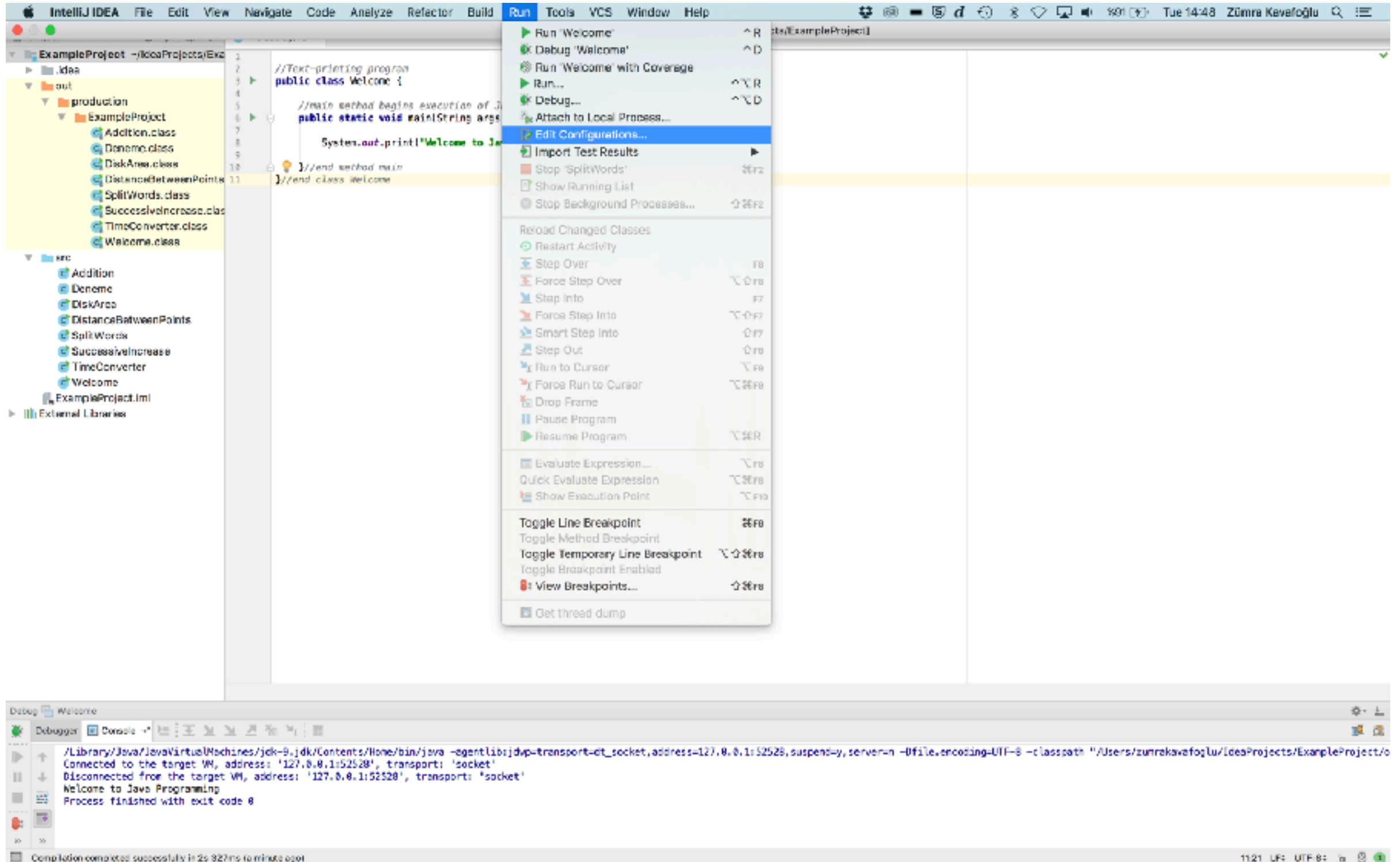
Java ile ilk program : Debug ile programı çalıştırmak



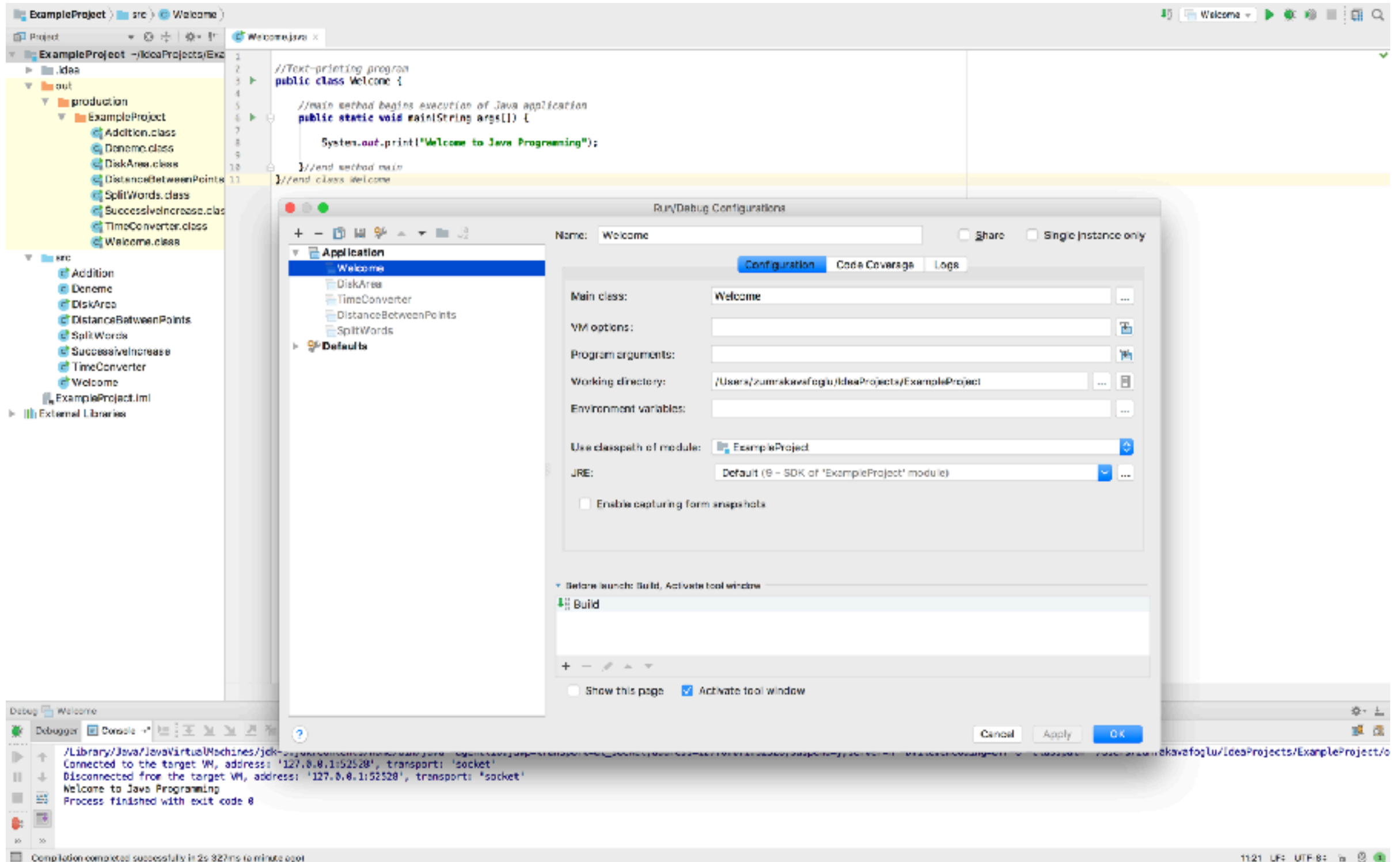
Java ile ilk program : Debug ile programı çalıştırmak



Java ile ilk program : Sınıf konfigürasyonlarını düzenlemek



Java ile ilk program : Sınıf konfigürasyonlarını düzenlemek



Java ile ilk program

Problem: Komut penceresine bir mesaj yazdırmak

Neler öğreneceğiz:

- Yorum satırları
- Sınıf deklarasyonu
- İsimlendirme kuralları
- Anahtar sözcükler
- main metodu
- İfadeler
- Derleme Hatası
- print ve println metodları
- Özel karakterler

Java ile ilk program : Ekrana bir metin satırı yazdırmak

```
//Text-printing program
public class Welcome {

    public static void main(String args[]){
        //main method begins execution of Java application
        System.out.println("Welcome to Java Programming");
    }//end method main
}//end class Welcome 1
```

Çıktı

Welcome to Java Programming!

Java ile ilk program : Yorum Satırları

```
//Text-printing program
public class Welcome {

    public static void main(String args[]){
        //main method begins execution of Java application
        System.out.println("Welcome to Java Programming");
    }//end method main
}//end class Welcome 1
```

- Bir java programında yorumlar kod hakkında açıklamalar yazmak için kullanılır.
- Java derleyicisi yorumları yok sayar.
- Tek satırlık yorumlar:
//Bu bir yorumdur.
- Birden fazla satırlı yorumlar:
/* Bunlar yorumlardır
Bunlar yorumlardır */
- Her programın başına o programın ne yaptığını yorum olarak not etmeyi alışkanlık haline getirmelisiniz.
- Her sınıf deklarasyonunun başına adınız, tarihi ve sınıfın genel görevini açıklamaya çalışın.
-

Java ile ilk program : Sınıf ismi

```
//Text-printing program
public class Welcome {

    public static void main(String args[]){
        //main method begins execution of Java application
        System.out.println("Welcome to Java Programming");
    }//end method main
}//end class Welcome 1
```

İsimlendirme Kuralları

- İsmi yazılan program hakkında fikir vermesi gerekir.
- İlk karakter bir rakam olamaz.
- İsim sadece harflerden, rakamlardan, alt çizgi _ ve dolar \$ işaretlerinden oluşabilir.
- İsim genellikle büyük harfle başlar ve isimdeki her bir sözcük büyük harfle başlar. (ExampleProgram)
- İsimde boşluk olmaz.
- İsim javanın anahtar sözcüklerinden biri olamaz.
- Maksimum karakter sayısı sınırsızdır ancak genelde mümkün olduğunca az karakter kullanılmalıdır.
- İsimlerde Türkçe karakterler kullanılmaz.
- Java büyük harf - küçük harf duyarlıdır(case sensitive).

Java'nın anahtar sözcükleri

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

Bazı yanlış isimler

2ab3 rakamla başlıyor

e*6 * işareti içeriyor

while java anahtar sözcüğü

Java ile ilk program : Sınıf deklarasyonu

```
1
2  //Text-printing program
3 ▶ public class Welcome {
4
5     //main method begins execution of Java application
6 ▶     public static void main(String args[]) {
7
8         System.out.println("Welcome to Java Programming");
9
10        }//end method main
11    }//end class Welcome
12
```

- Her sınıfın ismini içeren bir deklarasyonu yapılmalıdır.
- Dosya adı sınıf adıyla aynı olmalıdır ve dosyanın uzantısı .java olmalıdır. Örneğin bu sınıfı içeren dosyanın adı Welcome.java olmalıdır.

Java ile ilk program : Sınıf gövdesi ve parantezler

```
1
2  //Text-printing program
3  ▶ public class Welcome {
4
5      //main method begins execution of Java application
6  ▶  public static void main(String args[]) {
7
8      System.out.println("Welcome to Java Programming");
9
10     } //end method main
11 } //end class Welcome
12
```

↑
Sınıf gövdesi

- Sınıf gövdesi süslü parantez { ile açılmalı ve süslü parantez } ile kapatılmalıdır.

Java ile ilk program : main metodu

```
1
2  //Text-printing program
3  ▶ public class Welcome {
4
5      //main method begins execution of Java application
6  ▶  public static void main(String args[]) {
7
8      System.out.println("Welcome to Java Programming");
9
10     } //end method main
11 } //end class Welcome
12
```

Metod gövdesi

- Her java uygulaması içinde main metodu olan bir sınıf içermelidir.
- Uygulamalar main metoduyla çalışmaya başlar.
- main metodunun gövdesi { ile açılmalı ve } ile kapatılmalıdır.(Bu tüm metodlar için geçerlidir)
- Metodları ileriki derslerde detaylı olarak öğreneceğiz.
- Şimdilik her uygulamada yukarıdaki üstü taralı satırları kopyalayabilirsiniz.

Java ile ilk program : İfadeler

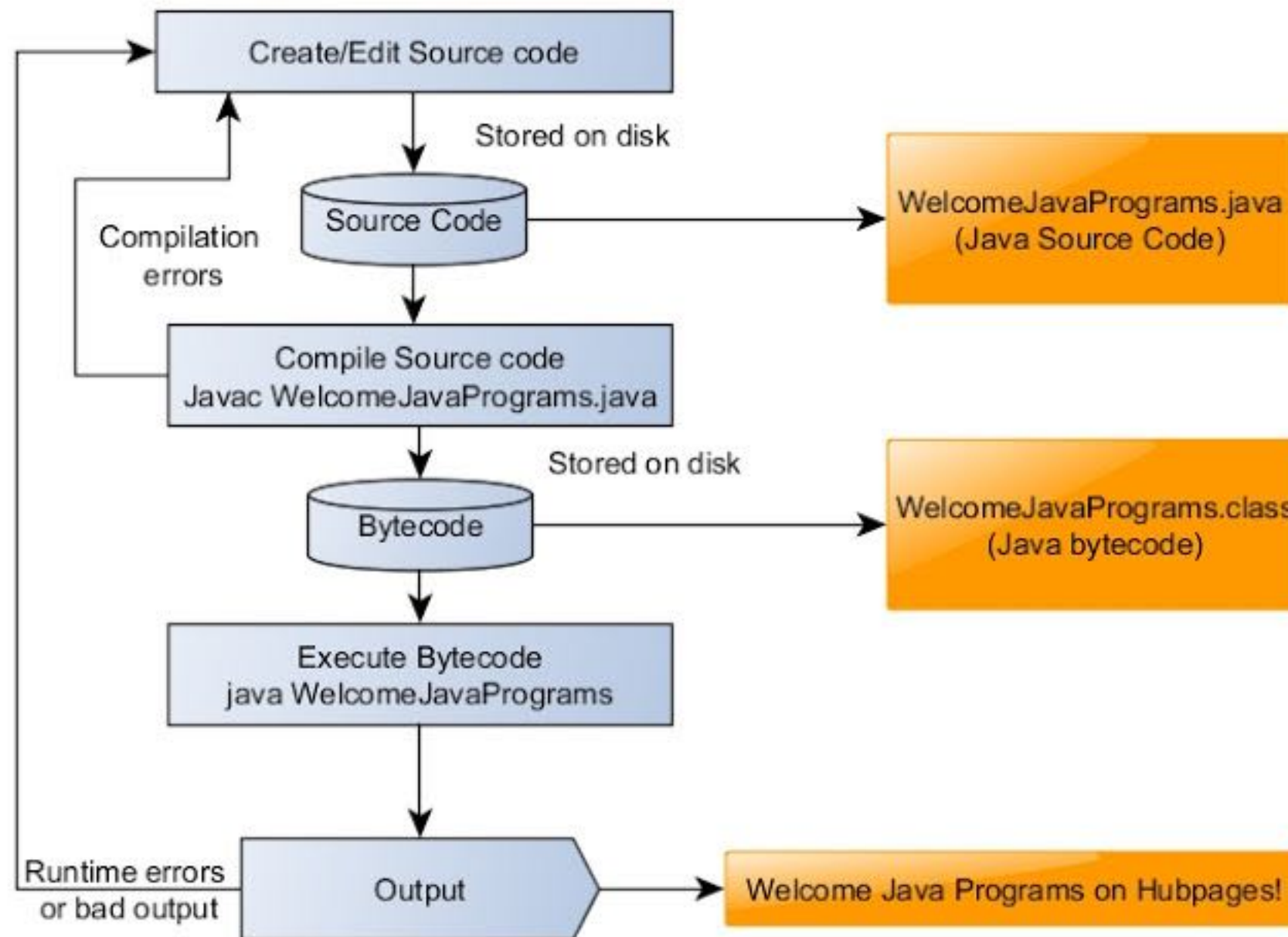
```
1
2  //Text-printing program
3 ▶ public class Welcome {
4
5      //main method begins execution of Java application
6 ▶      public static void main(String args[]) {
7
8          System.out.println("Welcome to Java Programming");
9
10         }//end method main
11     }//end class Welcome
12
```

- İfadeler(Statement) bilgisayara istenilen eylemi/işlemi yapması için verilen talimatlardır.
- İşaretli ifade ekrana (komut penceresine) çift tırnak içindeki metnin yazılması talimatını verir.
- Her ifade noktalı virgül ile bitmelidir.

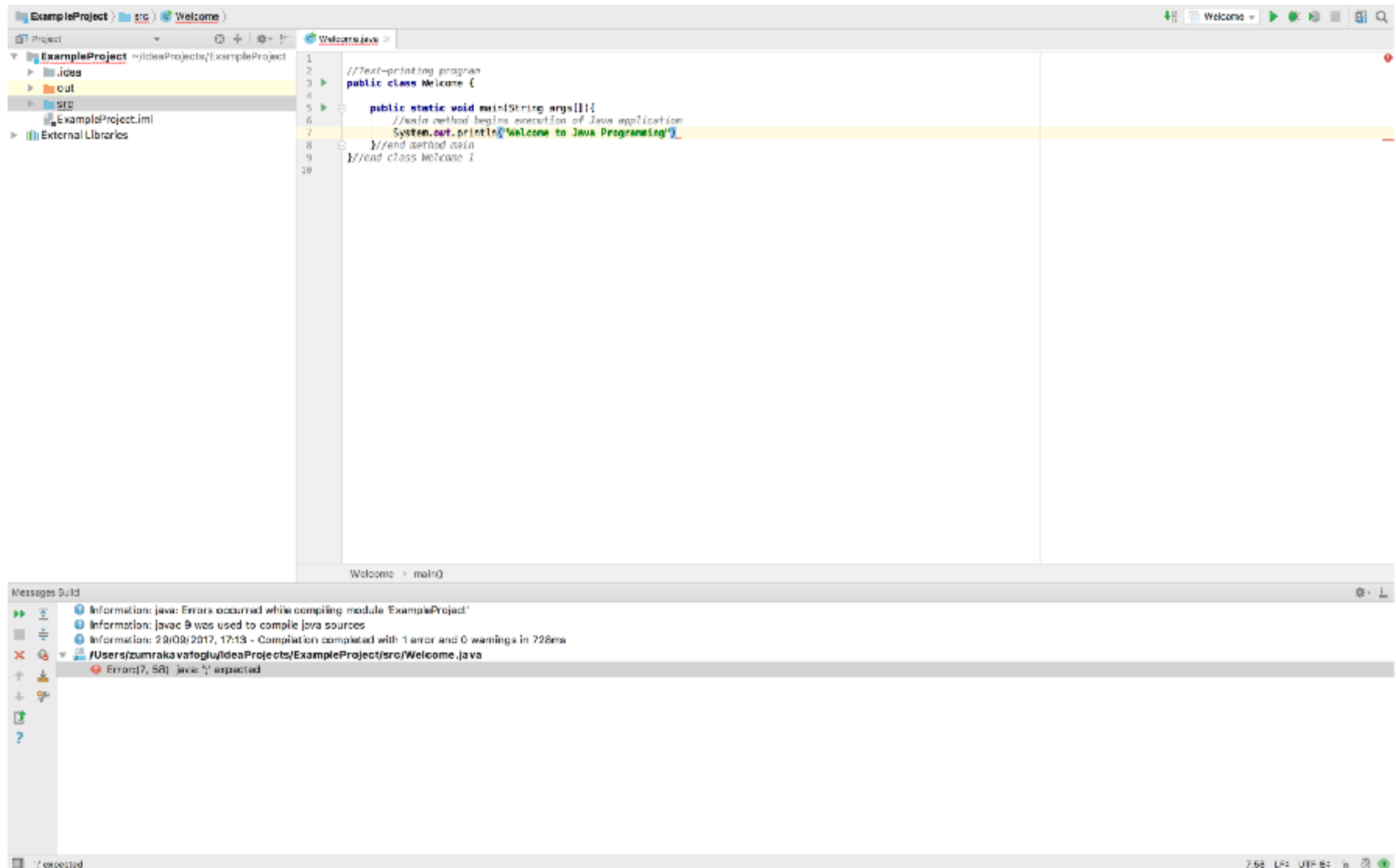
Derleme Hatası (Syntax Error / Compiler Error)

- Bir programlama dilinin syntax'ı o dilde doğru bir program yazmak için gerekli kuralları belirler.
- Derleyici Java'nın dil kurallarını(yani syntax'ini) ihlal eden bir kodla karşılaştığında derleme hatası ortaya çıkar.
- Bu durumda, derleyici bir .class dosyası üretmez, bunun yerine programcının yanlış kodu bulmasını ve düzeltmesine yardımcı olacak bir hata mesajı gösterir.
- Tüm derleme hataları düzeltilmeden program çalıştırılamaz.

Programı Derleme ve Çalıştırma



Derleme Hatası



Derleme Hatası

Program

```
1
2 //Text-printing program
3 public class Welcome {
4
5     public static void main(String args[]){
6         //main method begins execution of Java application
7         System.out.println("Welcome to Java Programming");
8     }//end method main
9 }//end class Welcome
10
```

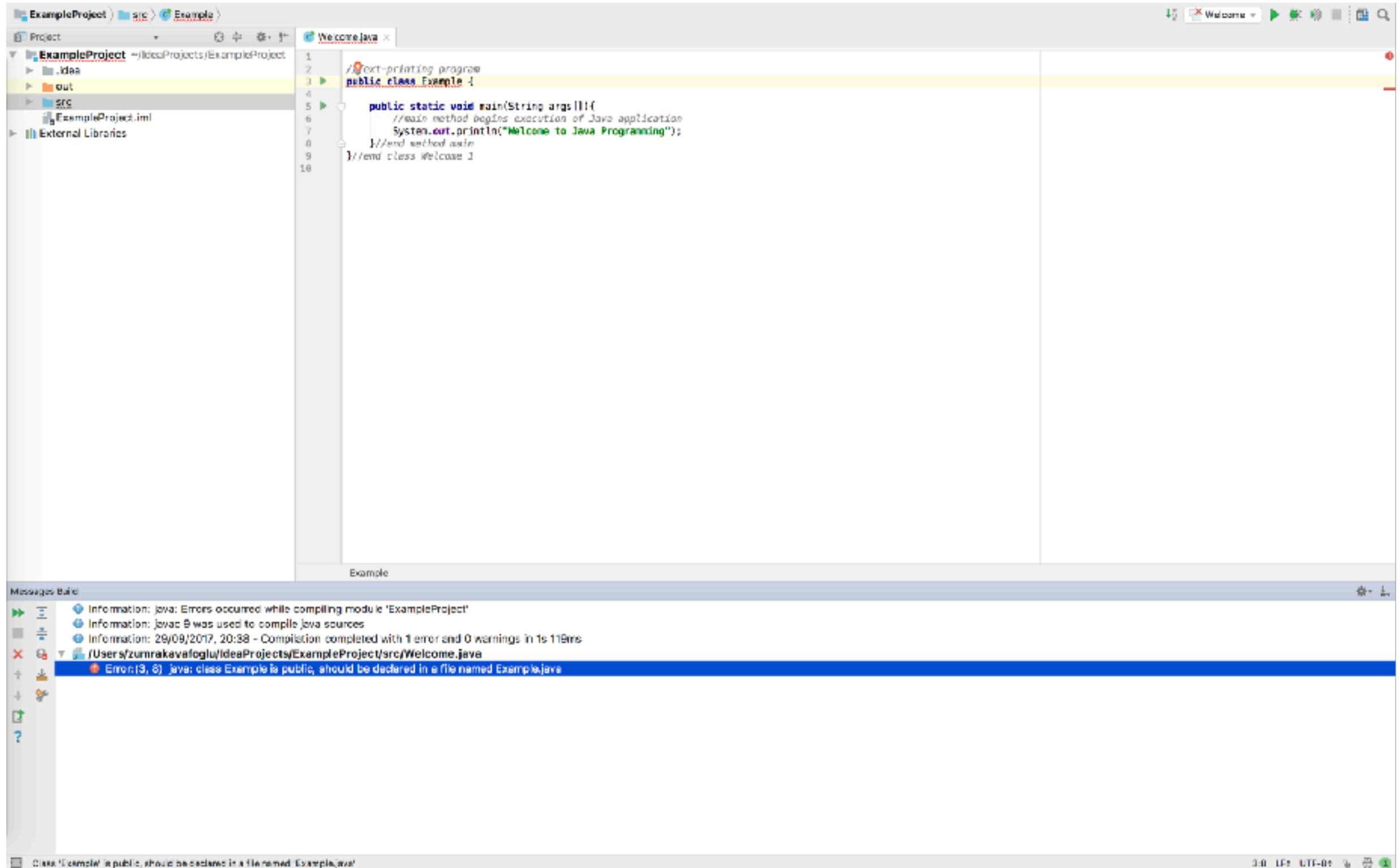
Hata Mesajı

- Information: java: Errors occurred while compiling module 'ExampleProject'
- Information: javac 9 was used to compile java sources
- Information: 29/09/2017, 17:13 - Compilation completed with 1 error and 0 warnings in 728ms

/Users/zumrakavafoglu/IdeaProjects/ExampleProject/src/Welcome.java

Error:(7, 58) java: ';' expected

Derleme Hatası



Derleme Hatası

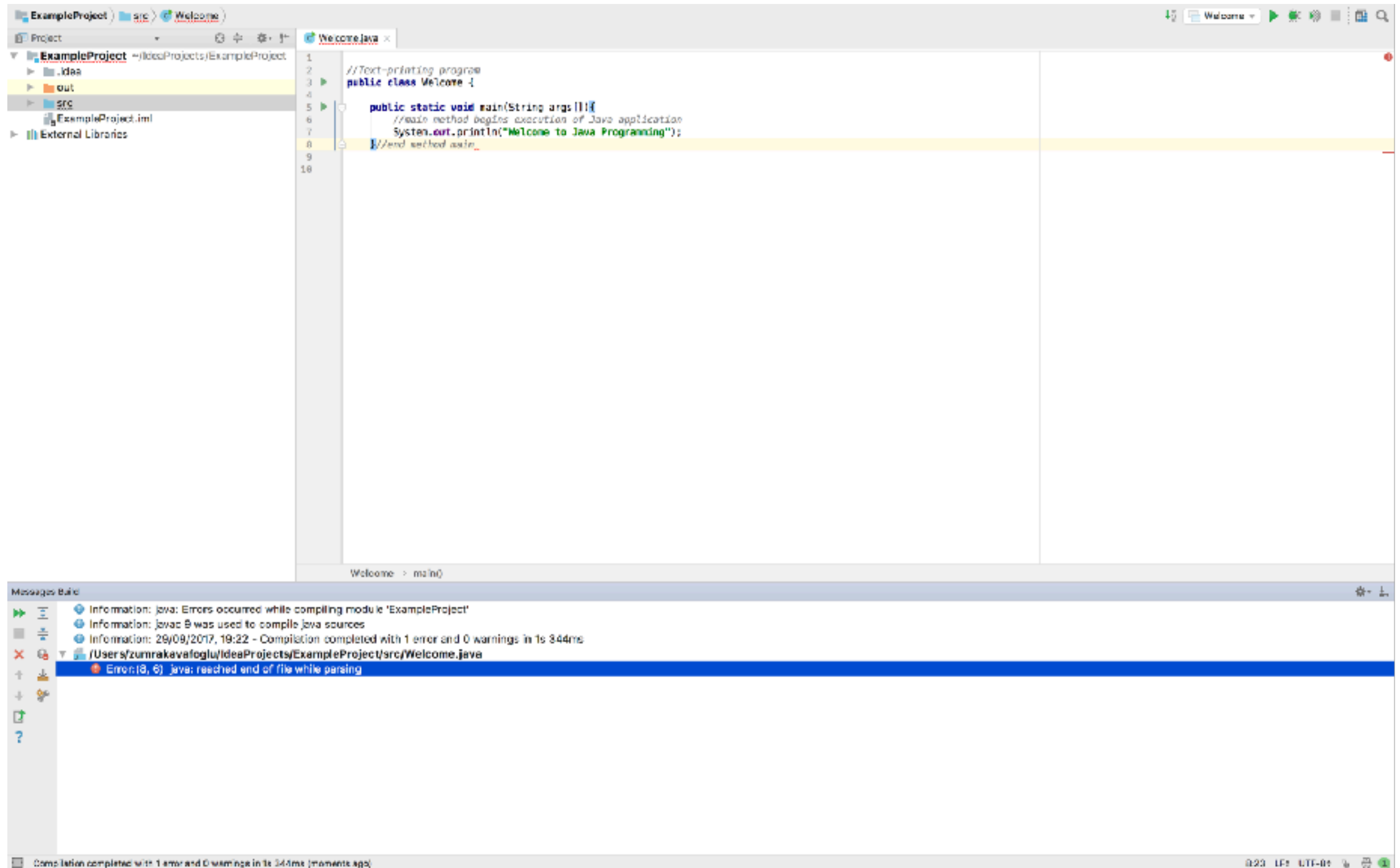
Program

```
Welcome.java x
1
2 //Text-printing program
3 public class Example {
4
5     public static void main(String args[]){
6         //main method begins execution of Java application
7         System.out.println("Welcome to Java Programming");
8     }//end method main
9 }//end class Welcome 1
10
```

Hata Mesajı

- Information: java: Errors occurred while compiling module 'ExampleProject'
 - Information: javac 9 was used to compile java sources
 - Information: 29/09/2017, 20:38 - Compilation completed with 1 error and 0 warnings in 1s 119ms
- ▼ /Users/zumrakavafoglu/IdeaProjects/ExampleProject/src/Welcome.java
- ! Error:(3, 8) java: class Example is public, should be declared in a file named Example.java

Derleme Hatası



Derleme Hatası

- Hata mesajı her zaman programdaki hatayı tam olarak ifade etmeyebilir. Ancak bu tip hatalarla karşılaştıkça hangi mesajın hangi hatadan doğmuş olabileceğini öğrenmeye başlayacaksınız.
- Programdaki hata her zaman gösterilen satırda olmayabilir.

Program

```
1
2 //Text-printing program
3 public class Welcome {
4
5     public static void main(String args[]){
6         //main method begins execution of Java application
7         System.out.println("Welcome to Java Programming");
8     }//end method main
9
```

Hata Mesajı

- Information: java: Errors occurred while compiling module 'ExampleProject'
- Information: javac 9 was used to compile java sources
- Information: 29/09/2017, 19:22 - Compilation completed with 1 error and 0 warnings in 1s 344ms

▼ /Users/zumrakavafoglu/IdeaProjects/ExampleProject/src/Welcome.java

❗ Error:(8, 6) java: reached end of file while parsing

print metodu

```
1
2 //Text-printing with multiple statements
3 ▶ public class Welcome2 {
4
5     //main method begins execution of Java application
6 ▶     public static void main(String args[]) {
7
8         System.out.print("Welcome");
9         System.out.println("to Java Programming");
10
11     } //end method main
12 } //end class Welcome
```

System.out.print imleci aynı satırda tutar, System.out.println imleci bir alt satıra indirir

Welcome to Java Programming!

Özel karakterler (Escape characters)

```
1
2 //Text-printing with multiple lines with a single statement
3 ▶ public class Welcome3 {
4
5     //main method begins execution of Java application
6 ▶     public static void main(String args[]) {
7
8         System.out.println("Welcome\nJava\nProgramming");
9
10    } //end method main
11 } //end class Welcome
```

Welcome
to
Java
Programming!

Her \n özel karakterinden sonra çıktıda yeni satıra geçilir.

Özel karakterler (Escape sequences)

Özel Karakter	Tanımı
<code>\n</code>	yeni satır
<code>\t</code>	tab
<code>\\</code>	backslash
<code>\'</code>	tek tırnak
<code>\"</code>	çift tırnak
<code>\b</code>	backspace

Tamsayı toplama programı

Problem: Değerleri verilen iki tamsayının toplamını ekrana yazdırınız.

Neler öğreneceğiz:

- Değişken tanımlama ve değer atama
- Primitif veri tipleri
 - Nümerik veri tipleri
 - Boolean
 - char
- String veri tipi
- İkili nümerik operatörler
- Kısayol atama operatörleri
- Arttırma ve azaltma operatörleri
- Math sınıfı metodları
- printf
- Hafızada değişkenler

Tamsayı toplama programı

```
Addition.java x
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
28 }
```

number1, number2 ve
sum değişkenlerini
tanımla

İyi değişken isimleri için öneriler

- Değişken isimleri küçük harfle başlar ve isimdeki takip eden her kelimenin baş harfi büyük yazılır
 - **örnek:** firstNumber
- Anlamlı değişken isimleri seçmek programın çok fazla yorum satırına veya dokümantasyona ihtiyaç olmadan anlaşılabilir olmasını sağlar. Değişken ismini kısa tutmak için anlamından feragat edilmemelidir.

Tamsayı toplama programı

Addition.java

```
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
```

number1 değişkenine
3 değerini ata

number2 değişkenine
8 değerini ata

Değişken tanımlama ve değer atama

Değişken tanımlama

int number1;

veri tipi

değişken adı

Değer atama

number1 = 3;

değişken adı

atanmak
istenen değer

atama operatörü
sağındaki değeri solundaki
değişkene atar

Değişken tanımlama ve değer atama

Değişken tanımlama

int number1;

veri tipi

değişken adı

Değer atama

number1 = 3;

değişken adı

atanmak
istenen değer

İlk değer ile tanımlama

int number1 = 3;

veri tipi

değişken adı

atanmak
istenen değer

Basit(Primitif) Veri Tipleri : Nümerik veri tipleri

- Basit (primitif) veri tipleri Java programlama dili tarafından önceden tanımlanmış veri tipleridir.
- Nümerik veri tipleri, boolean ve char primitif veri tipleridir.

Basit(Primitif) Veri Tipleri : Nümerik veri tipleri

isim	değer aralığı	depolama boyutu
byte	$-2^7(-128)$ to $2^7-1(127)$	8-bit signed
short	$-2^{15}(-32768)$ to $2^{15}-1(32767)$	16-bit signed
int	$-2^{31}(-2147483648)$ to $2^{31}-1(2147483647)$	32-bit signed
long	-2^{63} to $2^{63}-1$	64-bit signed
float	negatif aralık: $-3.4028235E+38$ to $-1.4E-45$ pozitif aralık: $1.4E-45$ to $3.4028235E+38$	32-bit IEEE 754
double	negatif aralık: $-1.7976931348623157E+308$ to $-4.9E-324$ pozitif aralık: $4.9E-324$ to $1.7976931348623157E+308$	64-bit IEEE 754

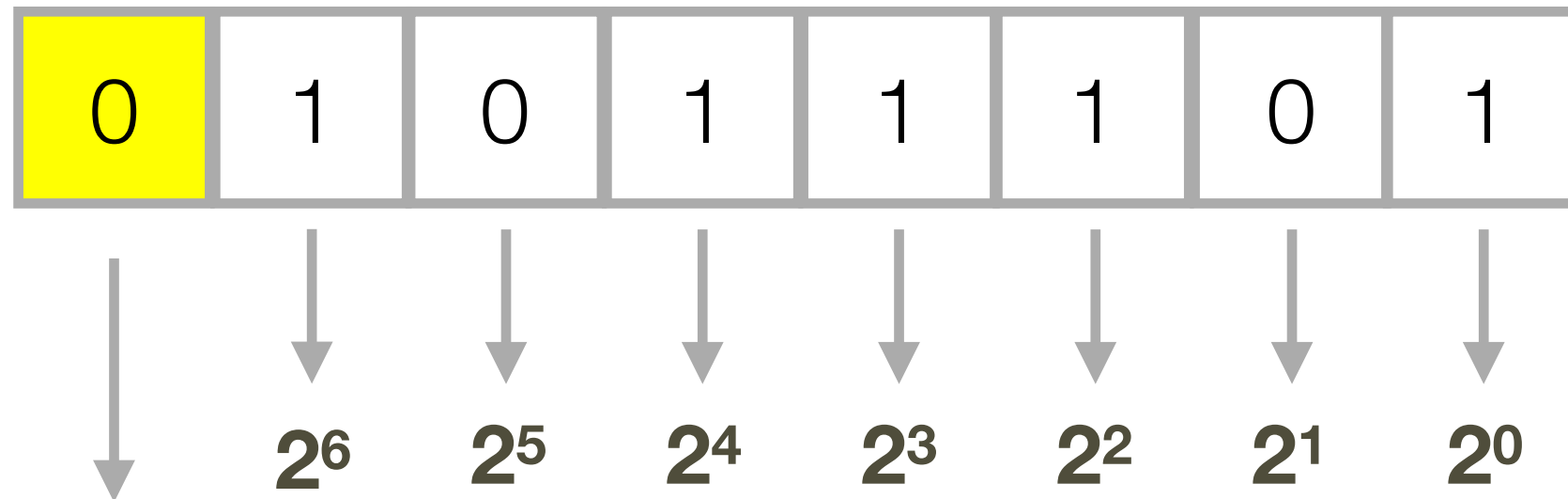
Nümerik veri tipleri : Tamsayılar

- **byte, short, int, long**: Hepsi tamsayı değişkenleri tanımlamak için kullanılır.
- Farkları hafızada kapladıkları yer ve saklayabilecekleri değerlerin büyüklüğüdür.

isim	değer aralığı	depolama boyutu
byte	$-2^7(-128)$ to $2^7-1(127)$	8-bit signed
short	$-2^{15}(-32768)$ to $2^{15}-1(32767)$	16-bit signed
int	$-2^{31}(-2147483648)$ to $2^{31}-1(2147483647)$	32-bit signed
long	-2^{63} to $2^{63}-1$	64-bit signed

Nümerik veri tipleri : Depolama

1 byte, 8 bit'e eşit ve her bit 1 veya 0 değerlerinden birini alıyor.



İşaret biti
pozitif : 0
negatif :1

$$\begin{aligned} &= 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 + 1 \cdot 2^3 + 1 \cdot 2^4 + 0 \cdot 2^5 + 1 \cdot 2^6 \\ &= 1 + 0 + 4 + 8 + 16 + 0 + 64 \\ &= 93 \end{aligned}$$

Nümerik veri tipleri : Rasyonel sayılar

- **double, float:** Rasyonel değişkenleri tanımlamak için kullanılır. Aralarındaki en önemli fark hafızada kapladıkları yer ve hassasiyetleridir.
- float değişkene değer atarken değerın sonuna f yazılmalıdır:

float x = 2.37f

isim	değer aralığı	depolama boyutu
float	negatif aralık: -3.4028235E+38 to -1.4E-45 pozitif aralık: 1.4E-45 to 3.4028235E+38	32-bit IEEE 754
double	negatif aralık: -1.7976931348623157E+308 to -4.9E-324 pozitif aralık: 4.9E-324 to 1.7976931348623157E+308	64-bit IEEE 754

Basit(Primitif) Veri Tipleri : boolean

boolean veri tipleri

- Mantıksal değerleri temsil etmek için vardır.
- İki farklı değer alabilir: true ve false

```
boolean var = true;
```

```
boolean var = false;
```

- boolean veri tiplerinin kullanımını ve boolean operatörleri bir sonraki ders detaylı öğreneceğiz.

Basit(Primitif) Veri Tipleri : char

Karakter değişken tipi:

- Alfabenin harfleri
- rakamlar
- + \$. , - ! gibi semboller

için kullanılır.

```
char letter = 'A';
```

Dizgi(String) veri tipi

- Karakter dizileridir.
- Primitif veri tipi değildir.

```
String message = "Hello World";
```

- Stringler birbirleriyle ve diğer veri tipleriyle + operatörü kullanılarak toplanabilirler.

```
int order = 1;  
  
String message = order+"1st message";  
  
System.out.println(message);
```

Çıktı

```
1st message
```

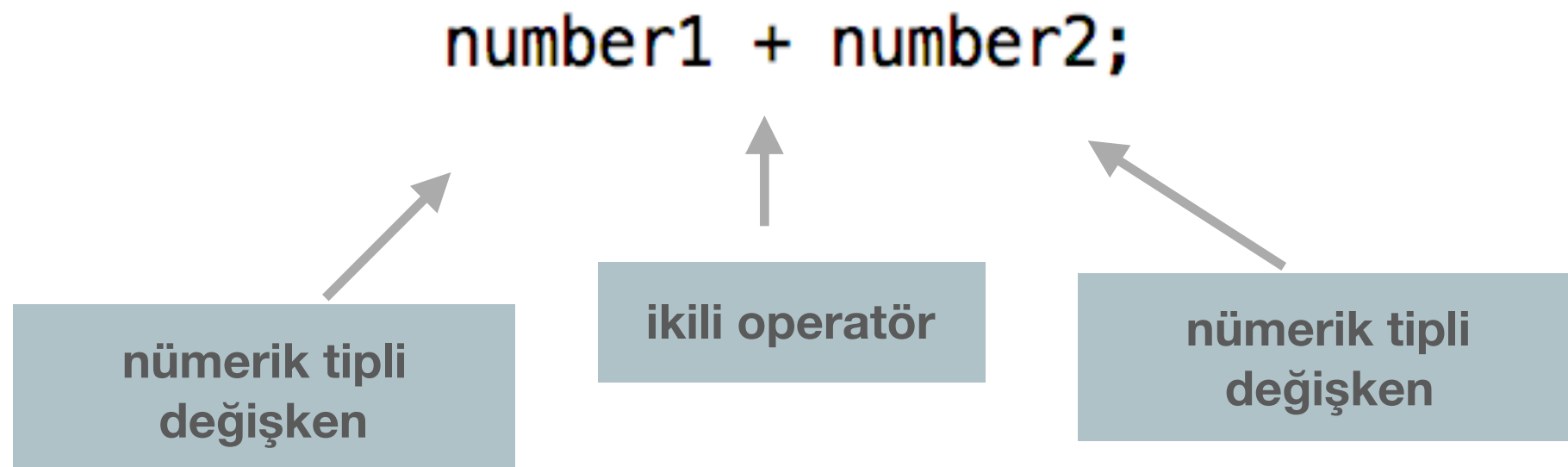
Tamsayı toplama programı

```
Addition.java x
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
28 }
```

number1 ve number2
değişkenlerinin toplamını
hesapla ve sum
değişkenine ata

İkili nümerik operatörler

İki nümerik tipli veriyi işleyen operatörlere ikili nümerik operatörler denir.



İkili nümerik operatörler

operatör	işlem
+	toplama
-	çıkarma
*	çarpma
/	bölme
%	kalan bulma

İkili nümerik operatörler : Tam sayı bölme

- int tipinde iki değişkenin bölme işleminin sonucu, bölüm tamsayı olmasa bile tamsayı çıkar.

```
int number1 = 7;  
int number2 = 2;  
int division = number1 / number2;  
System.out.print("Division is: " + division);
```

Çıktı

```
Division is: 3
```

İkili nümerik operatörler : Tam sayı bölme

Gerçek bölümü bulmak için **değişkenlerden en az biri** ve **sonuç tipi double** olarak tanımlanmalıdır.

```
double number1 = 7;  
  
int number2 = 2;  
  
double division = number1 / number2;  
  
System.out.println("Division is: " + division);
```

Çıktı

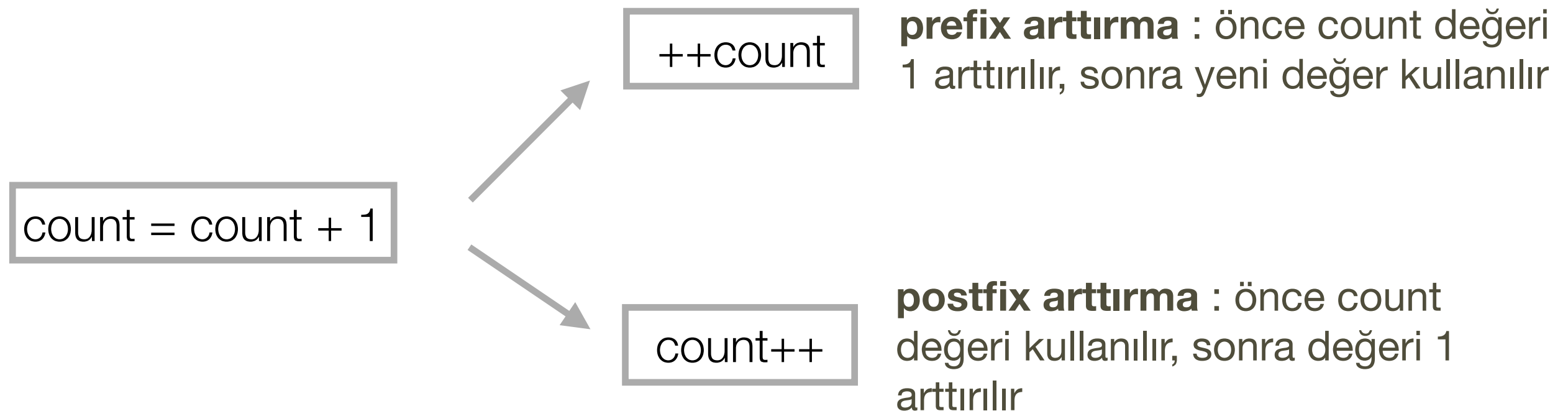
```
Division is: 3.5
```


Kısa yol atama operatörleri

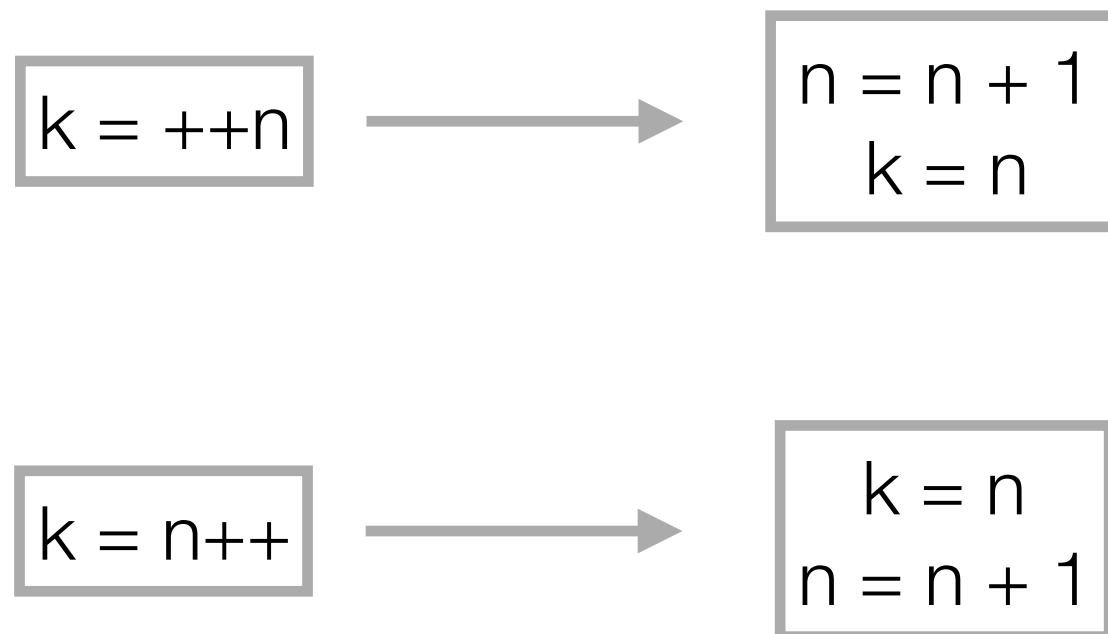
operatör	örnek	işlem
+=	i+=5	$i = i + 5$
-=	n -= 4	$n = n - 4$
=	k=8	$k = k * 8$
/=	t/=3	$t = t / 3$
%=	i %= 7	$i = i \% 7$

Arttırma ve Azaltma Operatörleri (Increment and Decrement Operators)

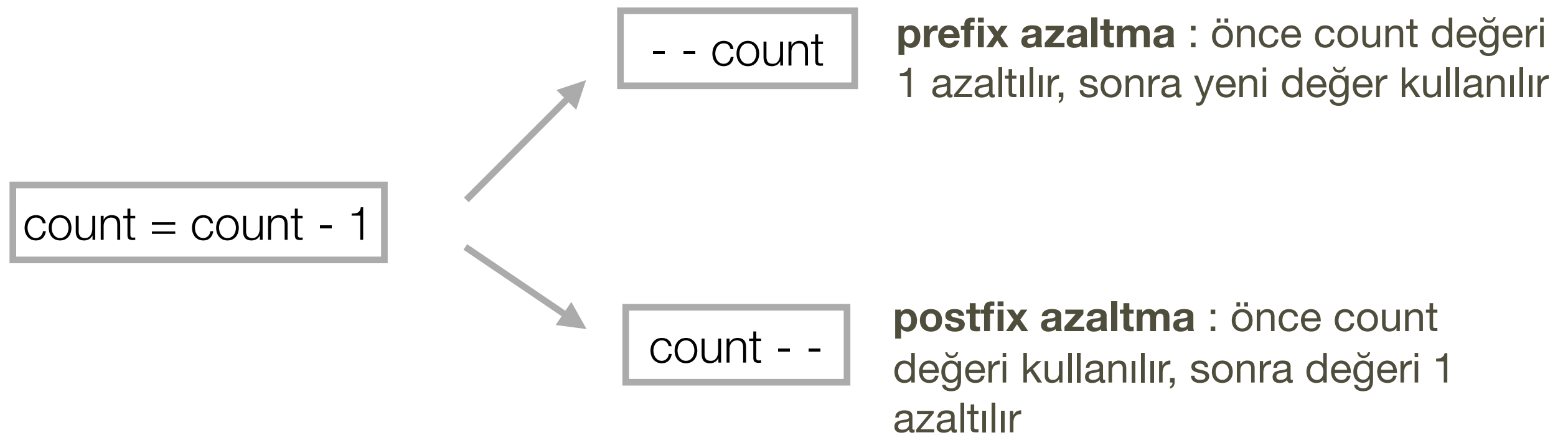
- Bir değişkenin 1 arttırıldığı ya da 1 azaltıldığı özel durumlar için arttırma ve azaltma işleçleri vardır.



Arttırma ve Azaltma Operatörleri (Increment and Decrement Operators)



Arttırma ve Azaltma Operatörleri (Increment and Decrement Operators)



Math sınıfı metodları

Method Name	Description	Returned Value
<code>abs(x)</code>	absolute value	same data type as argument
<code>pow(x1,x2)</code>	x1 raised to the x2 power	double
<code>sqrt(x)</code>	square root of x	double
<code>log(x)</code>	natural logarithm of x	double
<code>exp(x)</code>	e raised to the x power	double
<code>ceil(x)</code>	smallest integer value that is not less than x	double
<code>floor(x)</code>	largest integer value that is not greater than x	double
<code>min(x,y)</code>	smaller of its two arguments	same data type as arguments
<code>max(x,y)</code>	larger of its two arguments	same data type as arguments
<code>rint(x)</code>	closest integer value to the argument (in case of two closest integers, the even integer is returned)	double
<code>round(x)</code>	rounded value	integer
<code>random()</code>	random number between 0.0 inclusive and 1.0 exclusive	double
<code>sin(x)</code>	sine of x (x in radians)	double
<code>cos(x)</code>	cosine of x (x in radians)	double
<code>tan(x)</code>	tangent of x (x in radians)	double
<code>asin(x)</code>	arcsin of x	double
<code>acos(x)</code>	arccos of x	double
<code>atan(x)</code>	arctan of x	double

Math sınıfı metodları

Math.abs(-3)



3

Math.sqrt(16)



4

Math.min(2,7)



2

Tamsayı toplama programı

```
Addition.java x
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
28 }
```

Toplamı ekrana yazdır

Tamsayı toplama programı

```
Addition.java x
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
28 }
```

Sum is 11

Toplamı printf ile yazdırma

```
1 //Program for displaying the sum of two integers
2 public class Addition {
3     public static void main(String args[]) {
4         //first integer to add
5         int number1;
6
7         //second integer to add
8         int number2;
9
10        //sum of two integers
11        int sum;
12
13        //assign number1
14        number1 = 3;
15
16        //assign number2
17        number2 = 8;
18
19        //Compute the sum
20        sum = number1 + number2;
21
22        //Print the sum
23        System.out.printf("Sum is %d", sum);
24    }
25 }
```

printf veriyi istenen
formatta ekrana yazdırır

Sum is 11

Toplamı printf ile yazdırma

Yer tutucu argüman olarak verilen değişkenin metin içinde nerede yer alacağını ve formatını belirler.

```
System.out.printf("Sum is %d", sum);
```



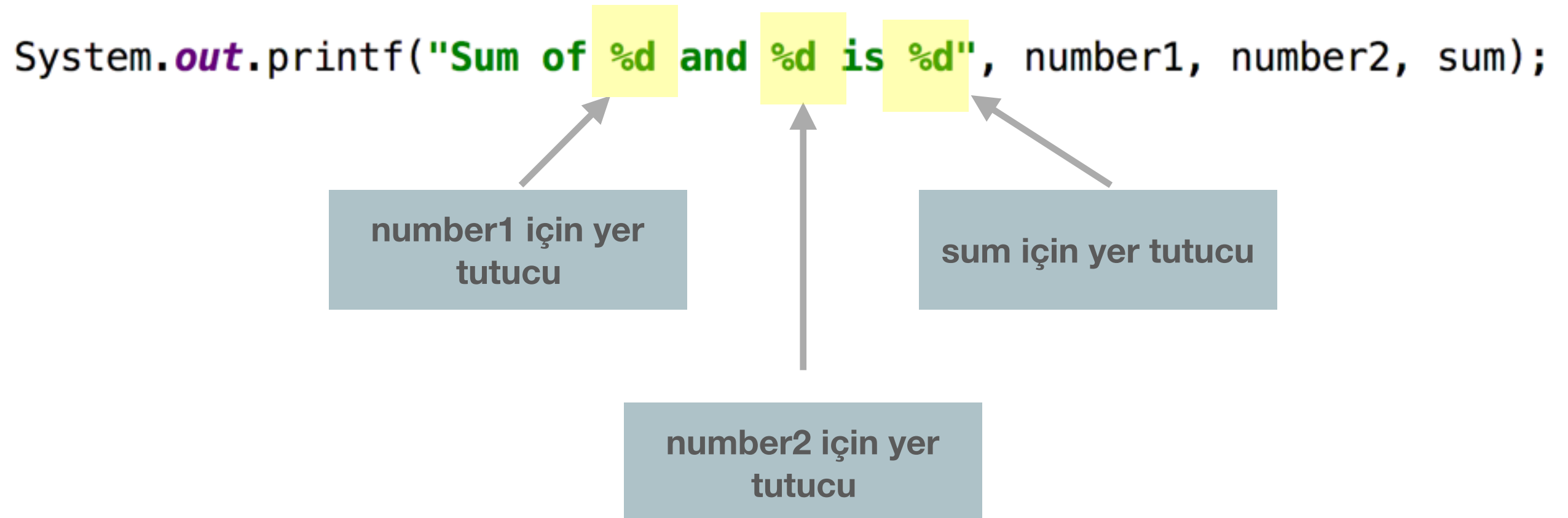
yer tutucu

değişken

printf: dönüştürücüler

dönüştürücü	
d	tamsayı
f	rasyonel sayı
c	karakter
s	string

Toplamı printf ile yazdırma

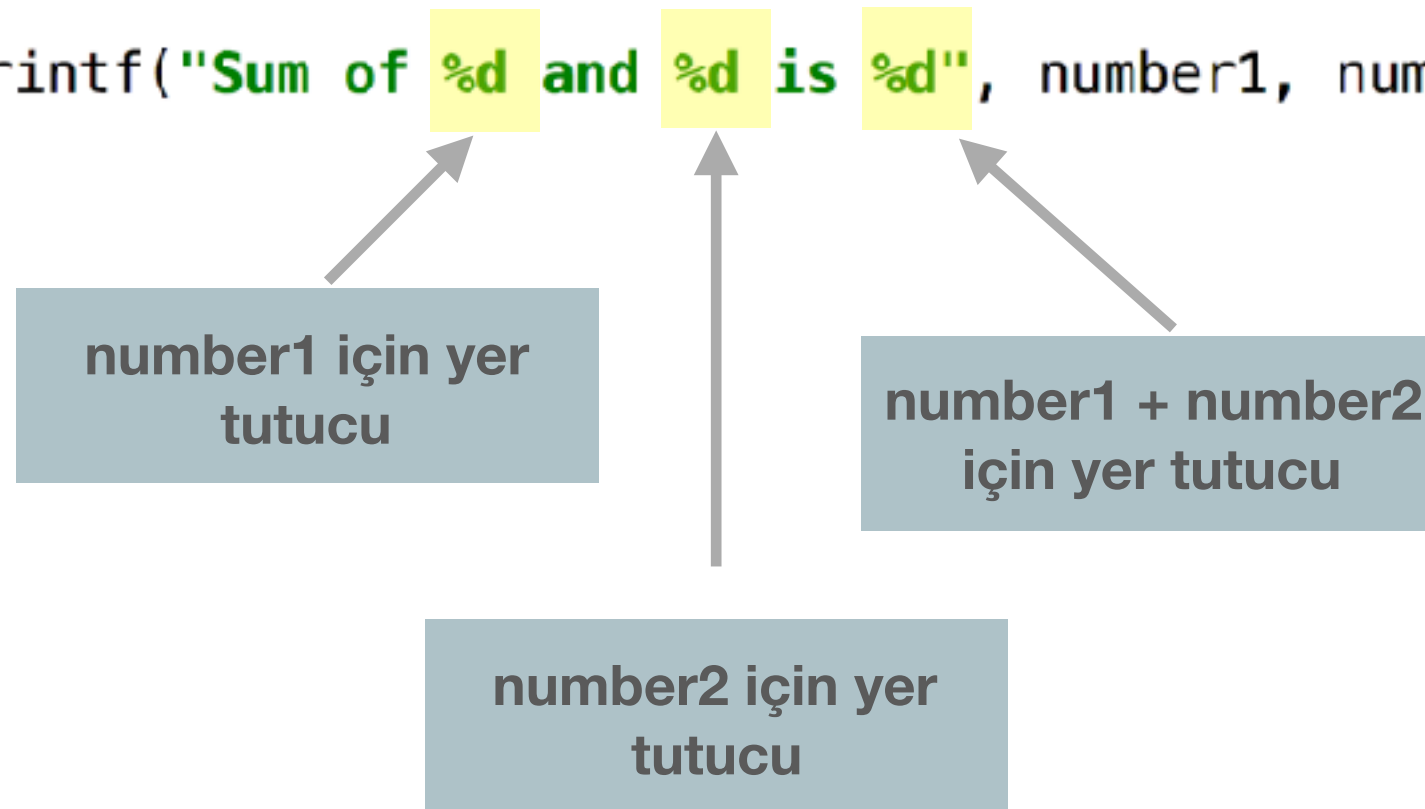


Çıktı

Sum of 3 and 8 is 11

printf: işlem printf'in içinde de yapılabilir

```
System.out.printf("Sum of %d and %d is %d", number1, number2, number1 + number2);
```



Çıktı

```
Sum of 3 and 8 is 11
```

printf: rasyonel sayıları formatlı yazdırma

```
double number = 12.367;  
System.out.printf("Number is %f", number);
```

Çıktı

```
Number is 12.367000
```



default olarak virgülden
sonra 6 haneli yazdırır

printf: rasyonel sayıları formatlı yazdırma

```
double number = 12.367;  
System.out.printf("Number is %.2f", number);
```

Çıktı

Number is 12.37



%.2f ile virgülden sonra 2
haneli yazdırır

Hafızada değişkenler

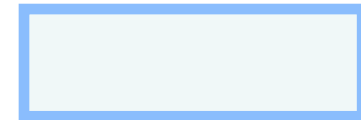
- Her değişkenin bir adı, tipi, boyutu ve değeri vardır.
- Değişkenin adı hafızada bir yere karşılık gelir.
- Bir değişkene yeni bir değer atandığında bir önceki değeri değiştirir(ve yok eder).
- Değişkenleri hafızadan okumak onların değerini değiştirmez.

Hafızada değişkenler

```
Addition.java x
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
28 }
```

number1 için hafızadan yer ayır

number1

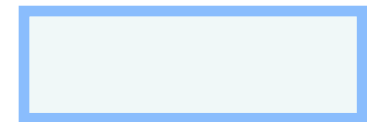


Hafızada değişkenler

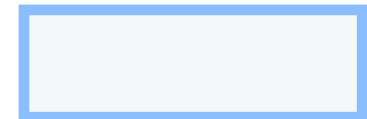
```
Addition.java x
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
28 }
```

number2 için hafızadan yer ayır

number1



number2



Hafızada değişkenler

```
Addition.java x
1 //Program for displaying the sum of two integers
2 public class Addition {
3     public static void main(String args[]) {
4         //first integer to add
5         int number1;
6         //second integer to add
7         int number2;
8         //sum of two integers
9         int sum;
10        //assign number1
11        number1 = 3;
12        //assign number2
13        number2 = 8;
14        //Compute the sum
15        sum = number1 + number2;
16        //Print the sum
17        System.out.print("Sum is " + sum);
18    }
19 }
```

sum için hafızadan yer ayır

number1

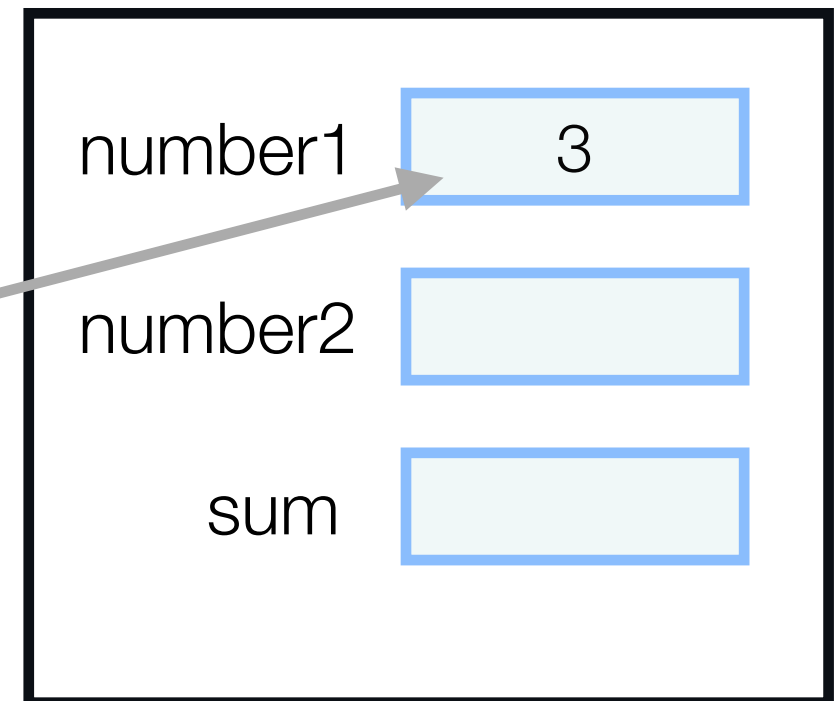
number2

sum

Hafızada değişkenler

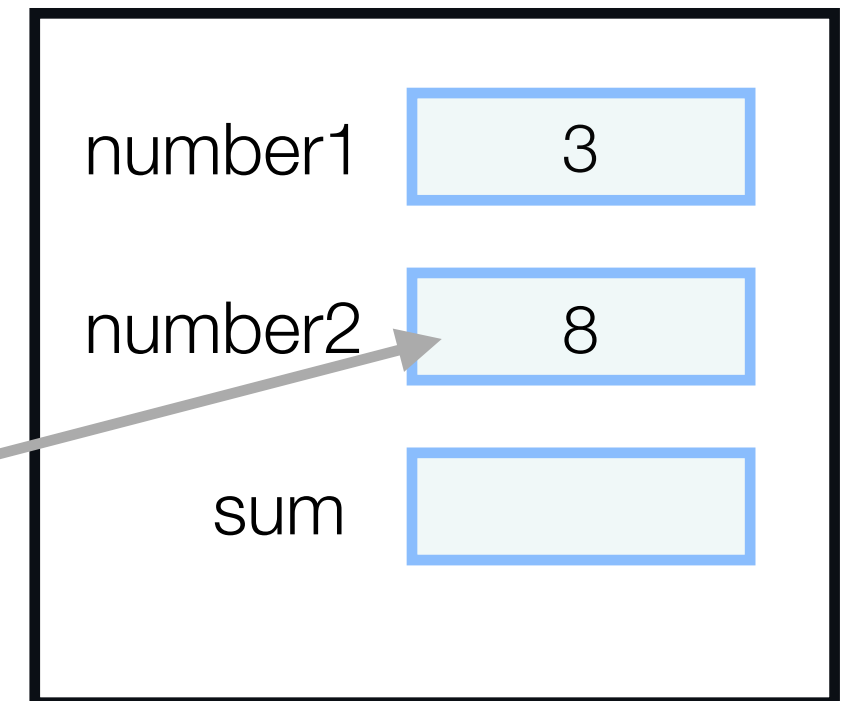
```
Addition.java x
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
28 }
```

3 değerini ata



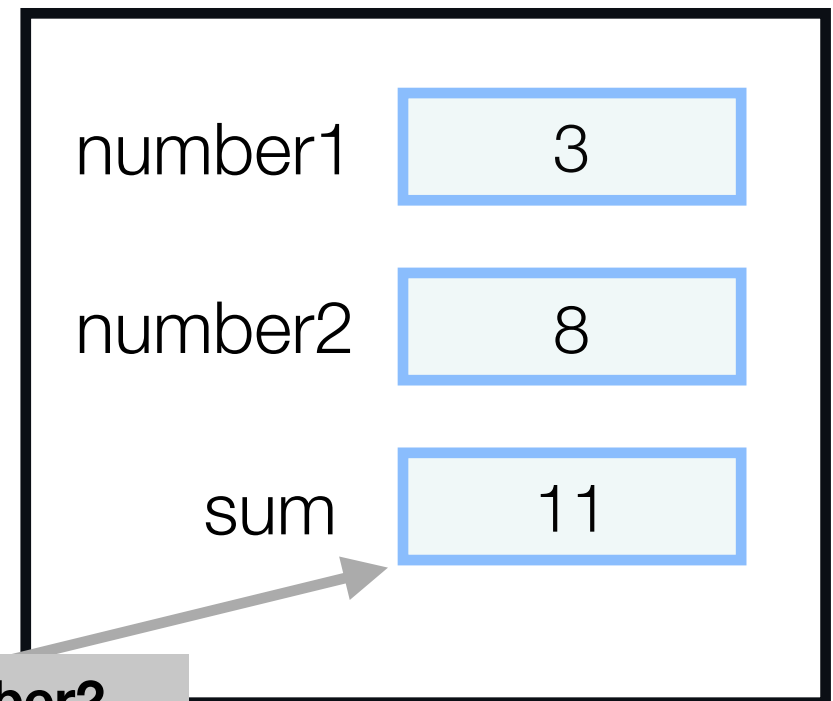
Hafızada değişkenler

```
Addition.java x
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
28 }
```



Hafızada değişkenler

```
Addition.java x
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
28 }
```



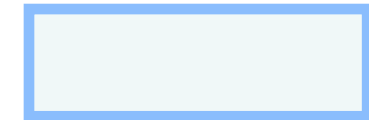
number1 + number2
değerini hesapla ve sum
değişkenine ata

Hafızada değişkenler

```
SuccessiveIncrease.java ×  
1  
2 //Program for increasing the value of  
3 // a number successively  
4 public class SuccessiveIncrease {  
5     public static void main(String args[]) {  
6  
7         int sum;  
8  
9         sum = 0;  
10  
11        sum = sum + 3;  
12  
13        sum += 7;  
14  
15        System.out.printf("Total sum is %d", sum);  
16    }  
17 }  
18  
19 }
```

sum için hafızadan yer ayır

sum



Hafızada değişkenler

```
SuccessiveIncrease.java ×  
1  
2 //Program for increasing the value of  
3 // a number successively  
4 public class SuccessiveIncrease {  
5     public static void main(String args[]) {  
6  
7         int sum;  
8  
9         sum = 0;  
10  
11        sum = sum + 3;  
12  
13        sum += 7;  
14  
15        System.out.printf("Total sum is %d", sum);  
16    }  
17 }  
18  
19 }
```

0 değerini ata

sum 0

Hafızada değişkenler

```
SuccessiveIncrease.java ×  
1  
2 //Program for increasing the value of  
3 // a number successively  
4 public class SuccessiveIncrease {  
5     public static void main(String args[]) {  
6  
7         int sum;  
8  
9         sum = 0;  
10  
11         sum = sum + 3;  
12  
13         sum += 7;  
14  
15         System.out.printf("Total sum is %d", sum);  
16  
17     }  
18  
19 }
```



3 değerini ata

sum = 0 + 3

Hafızada değişkenler

```
SuccessiveIncrease.java x
1
2 //Program for increasing the value of
3 // a number successively
4 public class SuccessiveIncrease {
5     public static void main(String args[]) {
6
7         int sum;
8
9         sum = 0;
10
11         sum = sum + 3;
12
13         sum += 7;
14
15         System.out.printf("Total sum is %d", sum);
16
17     }
18
19 }
```

sum = sum + 7

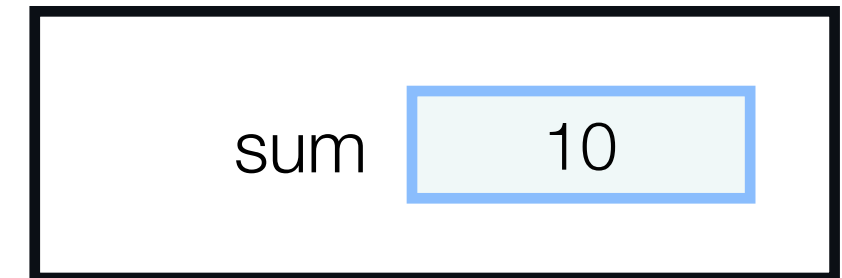
sum = 3 + 7

sum 10

10 değerini ata

Hafızada değişkenler

```
SuccessiveIncrease.java x
1
2 //Program for increasing the value of
3 // a number successively
4 public class SuccessiveIncrease {
5     public static void main(String args[]) {
6
7         int sum;
8
9         sum = 0;
10
11         sum = sum + 3;
12
13         sum += 7;
14
15         System.out.printf("Total sum is %d", sum);
16
17     }
18
19 }
```



Total sum is 10

Daire alanı hesaplama programı

Problem: Yarıçapı kullanıcı tarafından girilen dairenin alanını hesaplayıp, yarıçapıyla birlikte ekrana yazdırma.

Neler öğreneceğiz:

- final niteleyicisi
- Scanner sınıfıyla kullanıcı girdisi okuma

Daire alanı hesaplama programı

```
DiskArea.java x
1
2 import java.util.Scanner;
3
4 /*
5  Program that reads a radius from the user
6  and calculates the disk area with that radius
7  */
8
9 public class DiskArea {
10
11     public static void main(String args[]) {
12
13         //radius of the disk
14         double radius;
15
16         //area of the disk
17         double area;
18
19         //constant pi value
20         final double PI = 3.14;
21
22         //create Scanner to obtain input from command window
23         Scanner input = new Scanner(System.in);
24
25         System.out.print("Enter the radius: ");
26
27         //read radius from user
28         radius = input.nextDouble();
29
30         //calculate disk area
31         area = PI * radius * radius;
32
33         //Print radius and area to the command window
34         System.out.printf("\n Area of the disk with radius %.2f is %.2f ",radius,area);
35
36     }
```

final niteleyicisiyle
double tipinde sabit
PI değişkenini
tanımla

Daire alanı hesaplama programı

```
DiskArea.java x
1
2 import java.util.Scanner;
3
4 /*
5  Program that reads a radius from the user
6  and calculates the disk area with that radius
7  */
8
9 public class DiskArea {
10
11     public static void main(String args[]) {
12
13         //radius of the disk
14         double radius;
15
16         //area of the disk
17         double area;
18
19         //constant pi value
20         final double PI = 3.14;
21
22         //create Scanner to obtain input from command window
23         Scanner input = new Scanner(System.in);
24
25         System.out.print("Enter the radius: ");
26
27         //read radius from user
28         radius = input.nextDouble();
29
30         //calculate disk area
31         area = PI * radius * radius;
32
33         //Print radius and area to the command window
34         System.out.printf("\n Area of the disk with radius %.2f is %.2f ",radius,area);
35
36     }
```

*Scanner nesnesi
oluşturmak için bu
paketin import
edilmesi gerekli*

*Kullanıcıdan girdi
okumak için input
isimli Scanner
nesnesini tanımla*

Daire alanı hesaplama programı

```
DiskArea.java x
1
2 import java.util.Scanner;
3
4 /*
5  Program that reads a radius from the user
6  and calculates the disk area with that radius
7  */
8
9 public class DiskArea {
10
11     public static void main(String args[]) {
12
13         //radius of the disk
14         double radius;
15
16         //area of the disk
17         double area;
18
19         //constant pi value
20         final double PI = 3.14;
21
22         //create Scanner to obtain input from command window
23         Scanner input = new Scanner(System.in);
24
25         System.out.print("Enter the radius: ");
26
27         //read radius from user
28         radius = input.nextDouble();
29
30         //calculate disk area
31         area = PI * radius * radius;
32
33         //Print radius and area to the command window
34         System.out.printf("\n Area of the disk with radius %.2f is %.2f ",radius,area);
35
36     }
```

*Kullanıcıyı gireceği
değerle ilgili
bilgilendiren mesajı
yazdır*

Daire alanı hesaplama programı

```
DiskArea.java x
1
2 import java.util.Scanner;
3
4 /*
5  Program that reads a radius from the user
6  and calculates the disk area with that radius
7  */
8
9 public class DiskArea {
10
11     public static void main(String args[]) {
12
13         //radius of the disk
14         double radius;
15
16         //area of the disk
17         double area;
18
19         //constant pi value
20         final double PI = 3.14;
21
22         //create Scanner to obtain input from command window
23         Scanner input = new Scanner(System.in);
24
25         System.out.print("Enter the radius: ");
26
27         //read radius from user
28         radius = input.nextDouble();
29
30         //calculate disk area
31         area = PI * radius * radius;
32
33         //Print radius and area to the command window
34         System.out.printf("\n Area of the disk with radius %.2f is %.2f ",radius,area);
35
36     }
```

*Kullanıcının girdiği
double değeri radius
değişkenine ata*

Daire alanı hesaplama programı

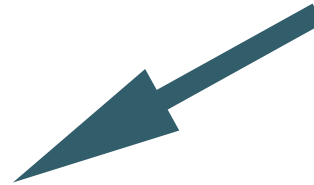
```
DiskArea.java x
1
2 import java.util.Scanner;
3
4 /*
5  Program that reads a radius from the user
6  and calculates the disk area with that radius
7  */
8
9 public class DiskArea {
10
11     public static void main(String args[]) {
12
13         //radius of the disk
14         double radius;
15
16         //area of the disk
17         double area;
18
19         //constant pi value
20         final double PI = 3.14;
21
22         //create Scanner to obtain input from command window
23         Scanner input = new Scanner(System.in);
24
25         System.out.print("Enter the radius: ");
26
27         //read radius from user
28         radius = input.nextDouble();
29
30         //calculate disk area
31         area = PI * radius * radius;
32
33         //Print radius and area to the command window
34         System.out.printf("\n Area of the disk with radius %.2f is %.2f ",radius,area);
35
36     }
```

Alanı hesaplayıp area
değişkenine ata

Daire alanı hesaplama programı

```
DiskArea.java x
1
2 import java.util.Scanner;
3
4 /*
5  Program that reads a radius from the user
6  and calculates the disk area with that radius
7  */
8
9 public class DiskArea {
10
11     public static void main(String args[]) {
12
13         //radius of the disk
14         double radius;
15
16         //area of the disk
17         double area;
18
19         //constant pi value
20         final double PI = 3.14;
21
22         //create Scanner to obtain input from command window
23         Scanner input = new Scanner(System.in);
24
25         System.out.print("Enter the radius: ");
26
27         //read radius from user
28         radius = input.nextDouble();
29
30         //calculate disk area
31         area = PI * radius * radius;
32
33         //Print radius and area to the command window
34         System.out.printf("\n Area of the disk with radius %.2f is %.2f ",radius,area);
35
36     }
```

*Yarıçapı ve alanı
ekrana yazdır*



Daire alanı hesaplama programı

1

Debugger Console



```
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/  
Connected to the target VM, address: '127.0.0.1:51435', tr  
Enter the radius:
```

Daire alanı hesaplama programı

1

Debugger Console →

```
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/  
Connected to the target VM, address: '127.0.0.1:51435', tr  
Enter the radius:
```

2

Debugger Console →

```
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/  
Connected to the target VM, address: '127.0.0.1:51435', tr  
Enter the radius: 4
```

Daire alanı hesaplama programı

1

Debugger Console

```
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/  
Connected to the target VM, address: '127.0.0.1:51435', tr  
Enter the radius:
```

2

Debugger Console

```
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/  
Connected to the target VM, address: '127.0.0.1:51435', tr  
Enter the radius: 4
```

3

Debugger Console

```
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/  
Connected to the target VM, address: '127.0.0.1:51460', tr  
Enter the radius: 4  
Disconnected from the target VM, address: '127.0.0.1:51460'
```

```
Area of the disk with radius 4.00 is 50.24  
Process finished with exit code 0
```

Scanner ile kullanıcıdan girdi okuma

```
Scanner input = new Scanner(System.in);  
radius = input.nextDouble();
```

- next(), nextByte(), nextShort(), nextInt(), nextLong(), nextFloat(), nextDouble() veya nextBoolean() metodlarını kullanarak byte, short, int, long, float, double veya boolean tipinde veri okuyabiliriz.

Scanner ile kullanıcıdan girdi okuma

```
Scanner input = new Scanner(System.in);  
radius = input.nextDouble();
```

- next(), nextByte(), nextShort(), nextInt(), nextLong(), nextFloat(), nextDouble() veya nextBoolean() metodlarını kullanarak byte, short, int, long, float, double veya boolean tipinde veri okuyabiliriz.

Farklı nümerik veri tipleri arasında işlemler

- İki değişken arasında yapılan işlemlerde
 - Eğer değişkenlerden biri double ise diğeri de double'a çevirilir.
 - Değil ise; eğer biri float ise diğeri de float'a çevirilir
 - Değil ise; eğer biri long ise diğeri de long'a çevirilir.
 - Değil ise ikisi de int'e çevirilir.

Veri tipi çevirme (Casting)

aralık genişler



byte, short, int, long, float, double

Veri tipi genişletme: Daha dar bir veri tipine sahip değişken daha geniş bir veri tipine otomatik olarak çevirilir.

```
int x = 2;
```

```
double y = x;
```

```
double a = 2.0;
```

```
int b = a;
```

Veri tipi çevirme (Casting)

aralık genişler



byte, short, int, long, float, double

Veri tipi genişletme: Daha dar bir veri tipine sahip değişken daha geniş bir veri tipine otomatik olarak çevirilir.



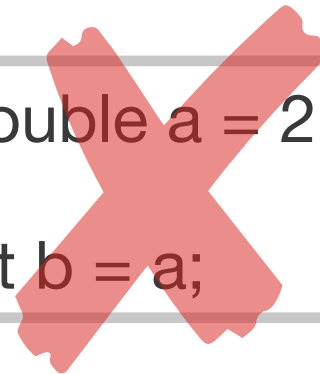
```
int x = 2;
```

```
double y = x;
```



```
double a = 2.0;
```

```
int b = a;
```



Veri tipi çevirme (casting)

aralık genişler



byte, short, int, long, float, double

Veri tipi daraltma:

~~double a = 2.0;
int b = a~~

double a = 2.0;
int b = (int)a;



int y = (int)4.8;



y 4

virgülden sonrası yok olur.