

Java Programlama, Veri Yapıları ve Bellek Yönetimi

4. Hafta

Dr. Öğr. Üyesi BÜŞRA ÖZDENİZCİ KÖSE

İşletme Bölümü

İşletme Fakültesi

Sayısal Veri Türleri: Gerçek Sayılar

- Java'da gerçek sayılar için iki önemli veri türü tanımlanmıştır: `float` ve `double`.
- `double` veri türü `float` veri türünün iki katı kadardır.
- Genellikle `double` veri türünü kullanmak daha iyi bir tercih olacaktır, doğru bir kesinlik (precision) sağlayacaktır.

	İsmi	Uzunluk	En Küçük Değer	En Büyük Değer
INTEGER (Tamsayı)	byte	1 byte	-2^7 (-128)	$2^7 - 1$ (+127)
	short	2 byte	-2^{15} (-32.768)	$2^{15} - 1$ (+32.767)
	int	4 byte	-2^{31} (-2.147.483.648)	$2^{31} - 1$ (+2.147.483.647)
	long	8 byte	-2^{63}	$2^{63} - 1$
FLOATING POINT (Gerçel Sayı)	float	4 byte	$-1,7 * 10^{38}$	$1,7 * 10^{38}$
	double	8 byte	$-3,4 * 10^{38}$	$3,4 * 10^{38}$
BOOLEAN (Mantıksal)	boolean	1 bit	true	false
CHARACTER (Karakter)	char	2 byte	0	$2^{16} - 1$ (+65.535)

Gerçek Sayılar

- Java programı içindeki bir sabit değeri **float** ya da **double** olarak somut şekilde ayrıştırmak istediğimizde, değerin sonuna float için { F, f }, double için ise { D, d } kullanabiliriz.
- Örneğin:
 - 100.2f ya da 100.2F olan float sabit değerini,
 - 100.2d ya da 100.2D olan double sabit değerini ifade eder.
- Gerçek sayılar, bilimsel format (exponential) kullanılarak da ifade edilebilir.
- Örneğin
 - 123.456 yerine 1.23456e+2 ya da 1.23456e2
 - 0.0123456 yerine 1.23456e-2 kullanılabilir

Gerçek değerler nasıl okunabilir?

Scanner Sınıf Metotları

Method	Data Type	Usage
<code>nextDouble()</code>	double	<code>double b = input.nextDouble();</code>
<code>nextFloat()</code>	float	<code>float s = input.nextFloat ();</code>

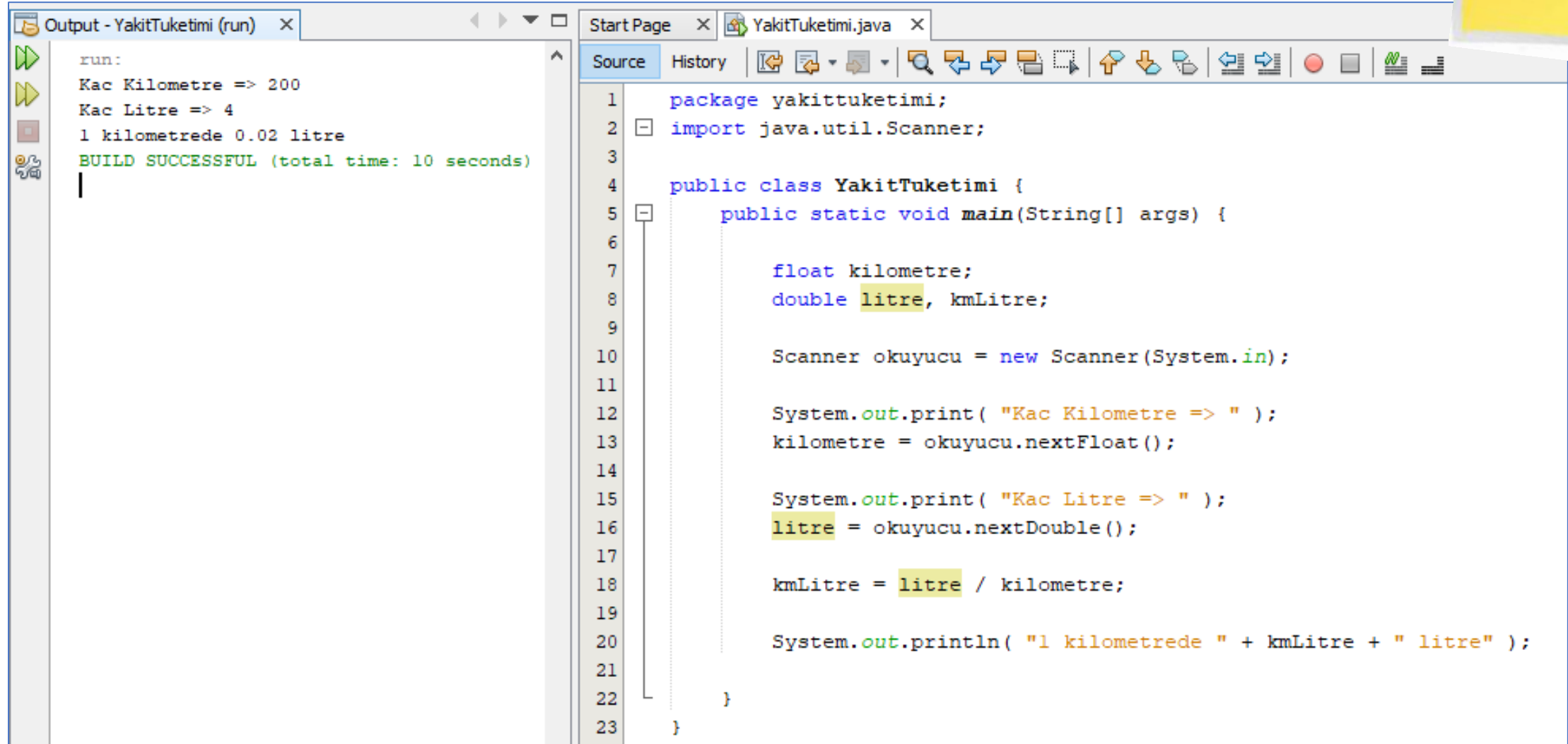
ComputeAverage.java



```
1  import java.util.Scanner; // Scanner is in the java.util package
2
3  public class ComputeAverage {
4      public static void main(String[] args) {
5          // Create a Scanner object
6          Scanner input = new Scanner(System.in);
7
8          // Prompt the user to enter three numbers
9          System.out.print("Enter three numbers: ");
10         double number1 = input.nextDouble();
11         double number2 = input.nextDouble();
12         double number3 = input.nextDouble();
13
14         // Compute average
15         double average = (number1 + number2 + number3) / 3;
16
17         // Display results
18         System.out.println("The average of " + number1 + " " + number2
19             + " " + number3 + " is " + average);
20     }
21 }
```

YakitTuketimi.java

DO IT
NOW!



The screenshot shows an IDE with two main panels. The left panel, titled 'Output - Yakittuketimi (run)', displays the program's execution output. The right panel, titled 'YakitTuketimi.java', shows the source code of the program. The code is a Java class named 'YakitTuketimi' that uses a 'Scanner' to read input from the user and calculate the fuel consumption rate in litres per kilometre.

```
run:
Kac Kilometre => 200
Kac Litre => 4
1 kilometrede 0.02 litre
BUILD SUCCESSFUL (total time: 10 seconds)

1 package yakittuketimi;
2 import java.util.Scanner;
3
4 public class Yakittuketimi {
5     public static void main(String[] args) {
6
7         float kilometre;
8         double litre, kmLitre;
9
10        Scanner okuyucu = new Scanner(System.in);
11
12        System.out.print( "Kac Kilometre => " );
13        kilometre = okuyucu.nextFloat();
14
15        System.out.print( "Kac Litre => " );
16        litre = okuyucu.nextDouble();
17
18        kmLitre = litre / kilometre;
19
20        System.out.println( "1 kilometrede " + kmLitre + " litre" );
21
22    }
23 }
```

Sabit Sayısal Değerler: `final`

- Programda kullandığımız bazı değişkenlerin sabit değerleri olabilir ve programda bu değişkenler tanımlandıktan sonra değerleri değişmez.
- Örneğin,
 - yıl 12 ay,
 - hafta 7 gün,
 - gün 24 saat,
 - saat 60 dakika
 - pi sayısı, karenin kenar sayısı ve benzeri bilgiler sabit sayısal değerlere sahiptir.
- Değişken değerinin program boyunca değişmemesi ve sabit kalması gereken durumlarda, **`final`** anahtar kelimesi kullanılır.

Sabit Değer

Java'da sabit değerler **büyük harf** olacak şekilde tanımlanırlar; birden çok kelimeden oluşması durumunda ise kelimeler alt çizgi ile birleştirilirler.

```
01 public class AlanHesapla{
02     public static void main( String[] args ) {
03         double yaricap, alan;
04         final double PI_DEGERI = 3.14;
05         yaricap = 20;
06         alan = yaricap * yaricap * PI_DEGERI;
07         System.out.println("Area of the circle is "+ alan);
08     }
09 }
```

final veri_tipi SABIT_DEGER_ISMI = deger;

Java'da İsimlendirme Kuralları - 1



- Java program yapısının kurallarına ek olarak yazılı olmayan kurallar da vardır.
- Bu kurallara uyulması, bir programcı tarafından yazılan programların diğerleri tarafından da kolaylıkla anlaşılmasına olanak sağlar.

Java'da İsimlendirme Kuralları - 2



- ASCII karakter tablosunda sadece İngilizce harflerin olması nedeniyle bu dilde olmayan {örneğin: ç, ğ, ı, ö, ş, ü, Ç, Ğ, İ, Ö, Ü} harfler kullanılamaz. Örneğin, yakın yerine yakın, ağaç yerine agac kullanılması gerekir.
- İlk karakter { A .. Z, a .. z, \$, _ } arasından seçilmelidir. Rakam ile başlayamaz. Sonra herhangi bir ASCII karakter gelebilir. İstedığınız uzunlukta olabilir.
- Java dilinde belirli amaçlar için kullanılan özel kelimeler (reserved words) bulunmaktadır. Bu kelimeler dilde tanımlı amaçlar dışında –örneğin bir değişken ismi olarak– kullanılamazlar. **true**, **false**, **null** değişken ismi olamaz.

Java'da İsimlendirme Kuralları - 3



- İsimlendirmede büyük harf ile küçük harf farklı şekilde değerlendirilir. Java küçük ve büyük harflere duyarlıdır. Dolayısı ile **toplam**, **TOPLAM**, **Toplam** üç ayrı değişken olarak tanımlanabilir.
- Tanımlayıcılar özellikle değişkenleri, sabit değerleri, metotları, sınıfları ve paketleri isimlendirmede kullanılır. Açıklayıcı ifadeler kullanmak gerekir.
- Kısaltmaları çok tercih etmemek gerekir. Örneğin; **numberOfStudents** tanımlayıcısı **numStuds**, **numOfStuds**, veya **numOfStudents** tanımlamalarından daha iyi olacaktır !

Java'da İsimlendirme Kuralları - 4



	Kural	Örnek
Paket	Hiyerarşik bir sırada önce daha büyük, giderek daha küçük birimler yazılır	<code>java.util</code>
Sınıf	Her biri büyük harf ile başlayıp küçük harf ile devam eden bir ya da daha fazla isim kelimesinden oluşmalıdır	<code>class Masa</code> <code>class KagitSepeti</code>
Proje	Sınıf ismine benzer şekilde oluşturulur	<code>BenimProjem</code>
Metot	İlk kelimesi tamamen küçük, diğer kelimeler ilk harfi büyük bir ya da daha fazla kelimedenden oluşmalı, bu kelimeler birleştğinde bir eylem bildirmelidir.	<code>calis()</code> <code>cokCalis()</code> <code>tembellikYapma()</code> <code>nextInt()</code> <code>nextDouble()</code>
Değişken	İlk kelimesi tamamen küçük, diğer kelimeler ise ilk harfi büyük bir ya da daha fazla kelimedenden oluşmalı, bu kelimeler birleştğinde bir isim bildirmelidir. Alışlagelmiş olan x, y, z, i, j, k, c, s dışında tek harfli değişkenlerden kaçınılmalıdır.	<code>int ilkSayi,</code> <code>i, sayi, adet,</code> <code>c, karakter;</code>
Sabit	Sabit değerleri olan değişkenlerin tamamı büyük harfli bir ya da daha fazla kelimedenden oluşmalı ve kelimeler birbirine altçizgi ile bağlanmalıdır.	<code>final int MAX = 10;</code> <code>final int MAX_YAS= 35;</code>

java.lang Package ve Math Class

- java.lang.Math sınıfı, temel üstel, logaritma, karekök ve trigonometrik fonksiyonlar gibi temel matematiksel ve sayısal işlemler gerçekleştirmek için gerekli metotları içerir.

Method name	Description
<code>Math.abs (value)</code>	absolute value
<code>Math.ceil (value)</code>	rounds up
<code>Math.floor (value)</code>	rounds down
<code>Math.log10 (value)</code>	logarithm, base 10
<code>Math.max (value1, value2)</code>	larger of two values
<code>Math.min (value1, value2)</code>	smaller of two values
<code>Math.pow (base, exp)</code>	<i>base</i> to the <i>exp</i> power
<code>Math.random ()</code>	random <code>double</code> between 0 and 1
<code>Math.round (value)</code>	nearest whole number
<code>Math.sqrt (value)</code>	square root
<code>Math.sin (value)</code> <code>Math.cos (value)</code> <code>Math.tan (value)</code>	sine/cosine/tangent of an angle in radians
<code>Math.toDegrees (value)</code> <code>Math.toRadians (value)</code>	convert degrees to radians and back

Constant	Description
<code>Math.E</code>	2.7182818...
<code>Math.PI</code>	3.1415926...

Math.pow() metodu & Math.sqrt() metodu

- Math.pow(a, b) metodu a^b hesaplamak için kullanılır.
 - Math.pow(a, b) şeklinde kullanılır.
 - Math.pow (2, 3) $\rightarrow a^b (2^3)$ sonucunu sağlayacaktır
 - System.out.println(Math.pow(2, 3));
- Math.sqrt (a) metodu a'nın karekökü için kullanılır: (\sqrt{a})
 - System.out.println(Math.sqrt(4));
 - System.out.println(Math.sqrt(9));

DO IT
NOW!

Output - Matematik (run) x

run:
Bir sayı giriniz: 25
25 değerinin karekökü 5.0
25 değerinin karekökü 5.0
BUILD SUCCESSFUL (total time: 4 seconds)

Start Page x Matematik.java x

Source History

```
1  
2 package matematik;  
3  
4 import java.util.Scanner;  
5  
6 public class Matematik {  
7  
8     public static void main(String[] args) {  
9  
10         Scanner busra = new Scanner(System.in);  
11  
12         System.out.print ( "Bir sayı giriniz: " );  
13  
14         int a = busra.nextInt();  
15  
16         double b = Math.sqrt(a); // karekökünü hesaplıyorum  
        System.out.println( a + " değerinin karekökü " + b ); // ekrana yazdır  
  
        System.out.println( a + " değerinin karekökü " + Math.sqrt(a) );  
        // ekrana yazdırırken hesapladım, alternatif yol  
    }  
}
```

Output - Matematik (run) x

run:
Bir sayı giriniz: 10
10 değerinin karekökü 3.1622776601683795
10 değerinin karekökü 3.1622776601683795
BUILD SUCCESSFUL (total time: 6 seconds)

DO IT
NOW!

The image shows an IDE window with a file named `Matematik.java`. The code is as follows:

```
1 package matematik;
2
3
4 import java.util.Scanner;
5
6 public class Matematik {
7
8     public static void main(String[] args) {
9
10         Scanner busra = new Scanner(System.in);
11
12         System.out.print ( "Bir sayı giriniz: " );
13         double a = busra.nextDouble();
14
15         System.out.print ( "Kaçınıcı kuvvetini almak istiyorsun: " );
16         double b = busra.nextDouble();
17
18         double c = Math.pow(a, b); // a'nın b üslünü hesaplıyorum
19
20         System.out.println( a + " değerinin " + b + ". kuvveti " + c ); // ekrana yazdır
21
22         System.out.println( a + " değerinin " + b + ". kuvveti " + Math.pow(a, b) );
23         // ekrana yazdırırken hesapladım, alternatif yol
24     }
25 }
```

Three output windows are shown, each displaying the results of a program run:

Output - Matematik (run)

```
run:
Bir sayı giriniz: 2
Kaçınıcı kuvvetini almak istiyorsun: 3
2.0 değerinin 3.0. kuvveti 8.0
2.0 değerinin 3.0. kuvveti 8.0
BUILD SUCCESSFUL (total time: 3 seconds)
```

Output - Matematik (run)

```
run:
Bir sayı giriniz: 4
Kaçınıcı kuvvetini almak istiyorsun: 0,5
4.0 değerinin 0.5. kuvveti 2.0
4.0 değerinin 0.5. kuvveti 2.0
BUILD SUCCESSFUL (total time: 4 seconds)
```

Output - Matematik (run)

```
run:
Bir sayı giriniz: 10
Kaçınıcı kuvvetini almak istiyorsun: -2
10.0 değerinin -2.0. kuvveti 0.01
10.0 değerinin -2.0. kuvveti 0.01
BUILD SUCCESSFUL (total time: 6 seconds)
```

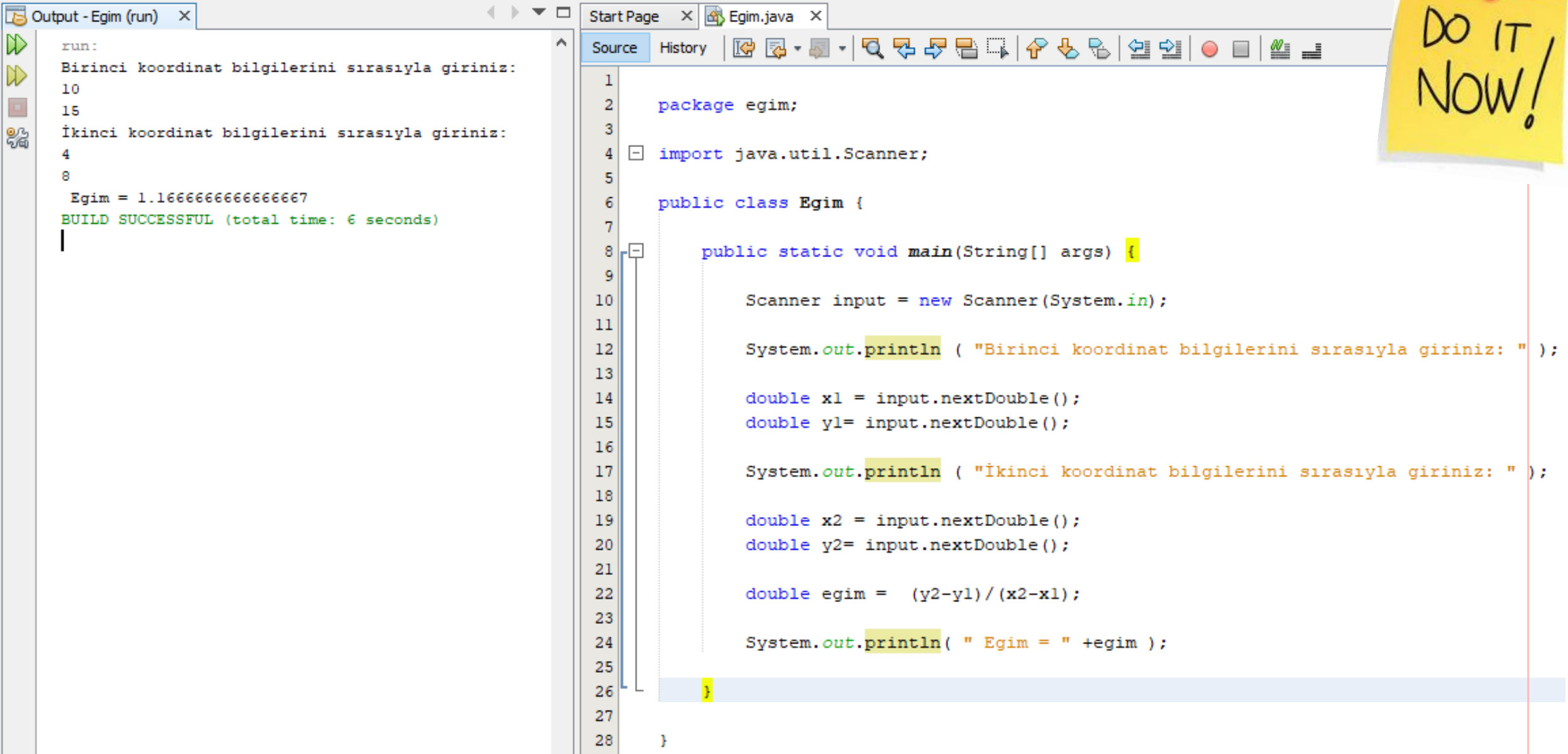
Egim.java



- İki noktanın koordinat bilgilerini kullanıcıdan alarak, iki nokta arasındaki eğimi hesaplayan bir Java programı geliştiriniz. Birinci noktanın koordinatları x1, y1 ve ikinci noktanın koordinatları x2, y2 olarak ifade edilebilir: $Eğim = (y2 - y1) / (x2 - x1)$

```
run:
Birinci koordinat bilgilerini sırasıyla giriniz:
4
5
İkinci koordinat bilgilerini sırasıyla giriniz:
3
8
Egim = -3.0
BUILD SUCCESSFUL (total time: 4 seconds)
```

DO IT
NOW!



The screenshot shows an IDE with two main panels. The left panel, titled 'Output - Egim (run)', displays the execution output of a Java program. The right panel, titled 'Egim.java', shows the source code of the program. The code is a Java class named 'Egim' that calculates the slope (egim) between two points entered by the user. The output shows the user entering coordinates (10, 15) and (4, 8), and the program calculating the slope as 1.1666666666666667. A yellow sticky note with the text 'DO IT NOW!' is pinned to the top right corner of the IDE window.

```
run:
Birinci koordinat bilgilerini sırasıyla giriniz:
10
15
İkinci koordinat bilgilerini sırasıyla giriniz:
4
8
Egim = 1.1666666666666667
BUILD SUCCESSFUL (total time: 6 seconds)

1
2 package egim;
3
4 import java.util.Scanner;
5
6 public class Egim {
7
8     public static void main(String[] args) {
9
10         Scanner input = new Scanner(System.in);
11
12         System.out.println ( "Birinci koordinat bilgilerini sırasıyla giriniz: " );
13
14         double x1 = input.nextDouble();
15         double y1= input.nextDouble();
16
17         System.out.println ( "İkinci koordinat bilgilerini sırasıyla giriniz: " );
18
19         double x2 = input.nextDouble();
20         double y2= input.nextDouble();
21
22         double egim = (y2-y1)/(x2-x1);
23
24         System.out.println( " Egim = " +egim );
25
26     }
27
28 }
```

Basamaklar.java



- Kullanıcının gireceği 3 basamaklı bir sayıyı basamaklarına ayırarak, basamakları ayrı satırlara ve basamakların toplamını ekrana yazdıran bir Java programı geliştiriniz.

```
Output - Basamaklar (run) X
run:
3 basamaklı bir sayı giriniz: 684
Yüzler basamağı: 6
Onlar basamağı: 8
Birler basamağı: 4
Basamaklar toplamı: 18
BUILD SUCCESSFUL (total time: 4 seconds)
```

YÜZLER BASAMAĞI	ONLAR BASAMAĞI	BİRLER BASAMAĞI
2	5	8



Output - Basamaklar (run) X

run:

3 basamaklı bir sayı giriniz: 985
Yüzler basamağı: 9
Onlar basamağı: 8
Birler basamağı: 5
Basamaklar toplamı: 22
BUILD SUCCESSFUL (total time: 4 seconds)

Start Page X Basamaklar.java X

Source History

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

```
package basamaklar;

import java.util.Scanner;

public class Basamaklar {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        System.out.print ( "3 basamaklı bir sayı giriniz: " );
        int sayi = input.nextInt();

        int a = sayi / 100; // yüzler basamağı

        int b = (sayi/10) % 10; //onlar basamağı

        int c = sayi % 10; //modunu alırsak birler basamağı

        System.out.println( "Yüzler basamağı: " +a);
        System.out.println( "Onlar basamağı: " +b);
        System.out.println( "Birler basamağı: " +c);

        System.out.println( "Basamaklar toplamı: " + (a+b+c) );

    }

}
```



NotOrtalama.java



- Not ortalaması hesaplayan bir Java programı geliştiriniz. Kullanıcının sınıf performans notunu, ara sınav notunu ve ardından final sınav notunu okuyarak not ortalaması hesaplamasını gerçekleştiriniz ve ekrana yazdırınız. Not ortalamasını oluşturan unsurların ağırlık yüzdeleri şu şekildedir: Sınıf performansı %30, Ara Sınav %30, Final sınavı %40.

```
run:
Sınıf performans notunuz => 60
Ara sınav notunuz => 80
Final sınav notunuz => 72
Dönem sonu notunuz 70
BUILD SUCCESSFUL (total time: 6 seconds)
```

Start Page x NotOrtalama.java x

Source History

5

6 public class NotOrtalama {

7

8 public static void main(String[] args) {

9

10 Scanner input = new Scanner(System.in);

11 double p, a, f, total;

12

13 System.out.print("Sınıf performans notunuz => ");

14 p = input.nextDouble();

15

16 System.out.print("Ara sınav notunuz => ");

17 a = input.nextDouble();

18

19 System.out.print("Final sınav notunuz => ");

20 f = input.nextDouble();

21

22 total = (p*30/100) + (a*30/100) + (f*40/100) ;

23

24 System.out.println("Dönem sonu notunuz = " + total);

25

26 }

27

28 }

notortalama.NotOrtalama > main >

Output - NotOrtalama (run) x

run:

Sınıf performans notunuz => 30

Ara sınav notunuz => 40

Final sınav notunuz => 60

Dönem sonu notunuz = 45.0

BUILD SUCCESSFUL (total time: 6 seconds)

DikdortgenPrizma.java



- Dikdörtgen bir prizmanın kenar uzunluklarını okuyup alanı ve hacmini hesaplayınız, sonra ekrana sonuçları yazdırınız !

```
01 Birinci kenarını giriniz => 3
02 İkinci kenarını giriniz => 4
03 Üçüncü kenarını giriniz => 5
04 Alan => 94
05 Hacim => 60
```


Output - DikdortgenPrizma (run) x

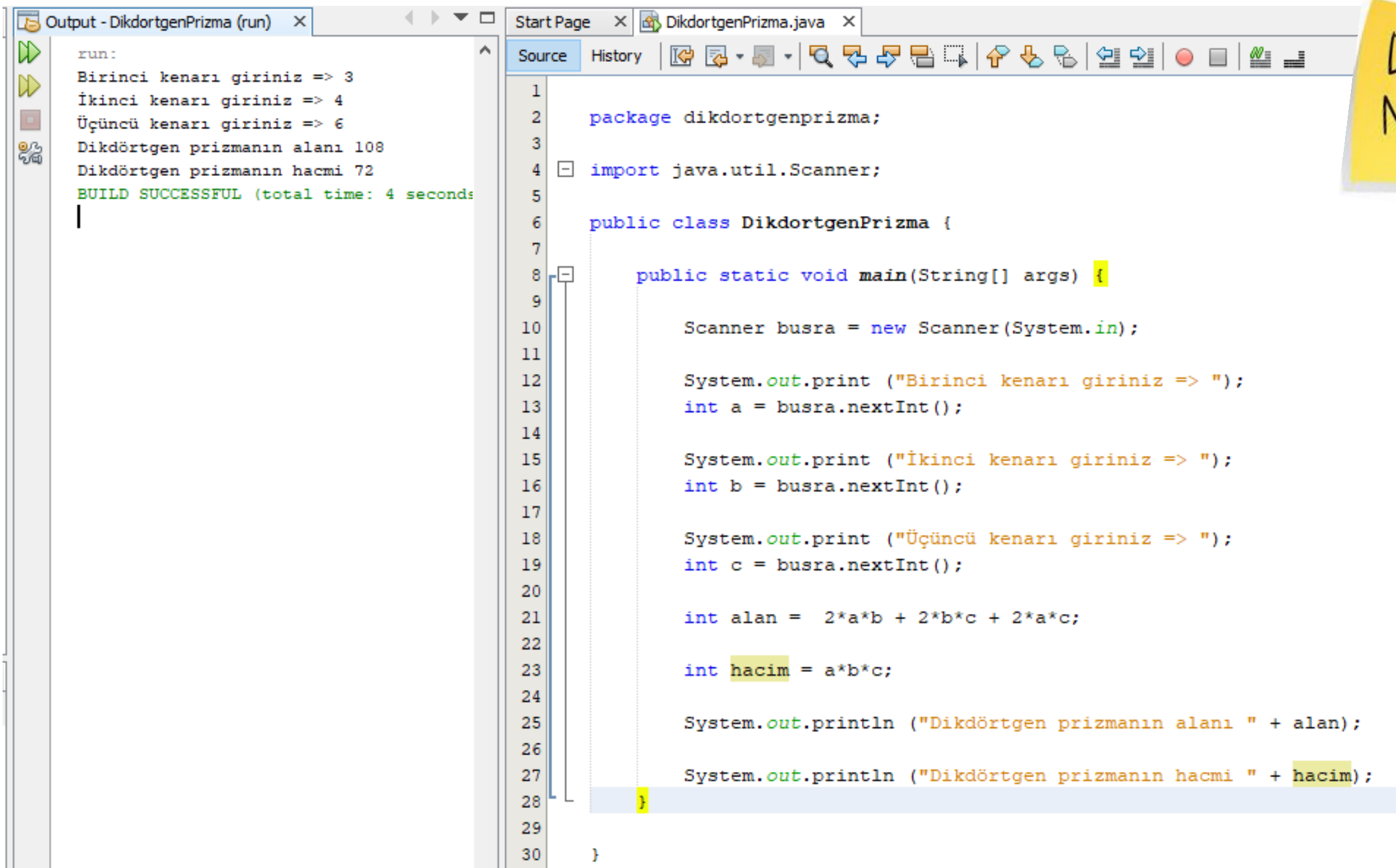
run:
Birinci kenarı giriniz => 3
İkinci kenarı giriniz => 4
Üçüncü kenarı giriniz => 5
Dikdörtgen prizmanın alanı 94
Dikdörtgen prizmanın hacmi 60
BUILD SUCCESSFUL (total time: 3 seconds)

Start Page x DikdortgenPrizma.java x

Source History

```
1 package dikdortgenprizma;  
2  
3  
4 import java.util.Scanner;  
5  
6 public class DikdortgenPrizma {  
7  
8     public static void main(String[] args) {  
9  
10         Scanner busra = new Scanner(System.in);  
11         int a, b, c;  
12         int alan, hacim;  
13         //tüm gerekli değişkenlerimi ve Scanner nesnemi tanımladım  
14  
15         //şimdi kullanıcıdan değerleri okuyalım ve değişkenlere atayalım  
16  
17         System.out.print ("Birinci kenarı giriniz => ");  
18         a = busra.nextInt();  
19  
20         System.out.print ("İkinci kenarı giriniz => ");  
21         b = busra.nextInt();  
22  
23         System.out.print ("Üçüncü kenarı giriniz => ");  
24         c = busra.nextInt();  
25  
26         alan = 2*a*b + 2*b*c + 2*a*c;  
27         hacim = a*b*c;  
28  
29         System.out.println ("Dikdörtgen prizmanın alanı " + alan);  
30  
31         System.out.println ("Dikdörtgen prizmanın hacmi " + hacim);  
32     }  
33  
34 }
```





```
Output - DikdortgenPrizma (run) x
run:
Birinci kenarı giriniz => 3
İkinci kenarı giriniz => 4
Üçüncü kenarı giriniz => 6
Dikdörtgen prizmanın alanı 108
Dikdörtgen prizmanın hacmi 72
BUILD SUCCESSFUL (total time: 4 seconds)

Start Page x DikdortgenPrizma.java x
Source History
1
2 package dikdortgenprizma;
3
4 import java.util.Scanner;
5
6 public class DikdortgenPrizma {
7
8     public static void main(String[] args) {
9
10         Scanner busra = new Scanner(System.in);
11
12         System.out.print ("Birinci kenarı giriniz => ");
13         int a = busra.nextInt();
14
15         System.out.print ("İkinci kenarı giriniz => ");
16         int b = busra.nextInt();
17
18         System.out.print ("Üçüncü kenarı giriniz => ");
19         int c = busra.nextInt();
20
21         int alan = 2*a*b + 2*b*c + 2*a*c;
22
23         int hacim = a*b*c;
24
25         System.out.println ("Dikdörtgen prizmanın alanı " + alan);
26
27         System.out.println ("Dikdörtgen prizmanın hacmi " + hacim);
28     }
29
30 }
```



Java'da Özel Anlamalı Kelimeler

abstract	continue	for	new	switch
assert	default	goto	package	synchronized
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

Any Questions?