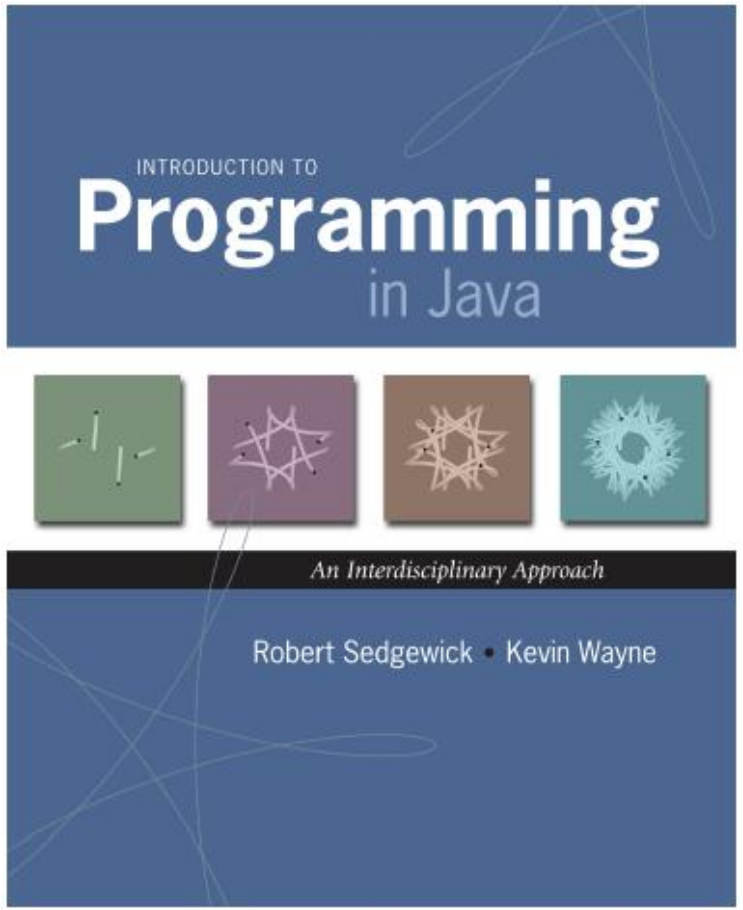
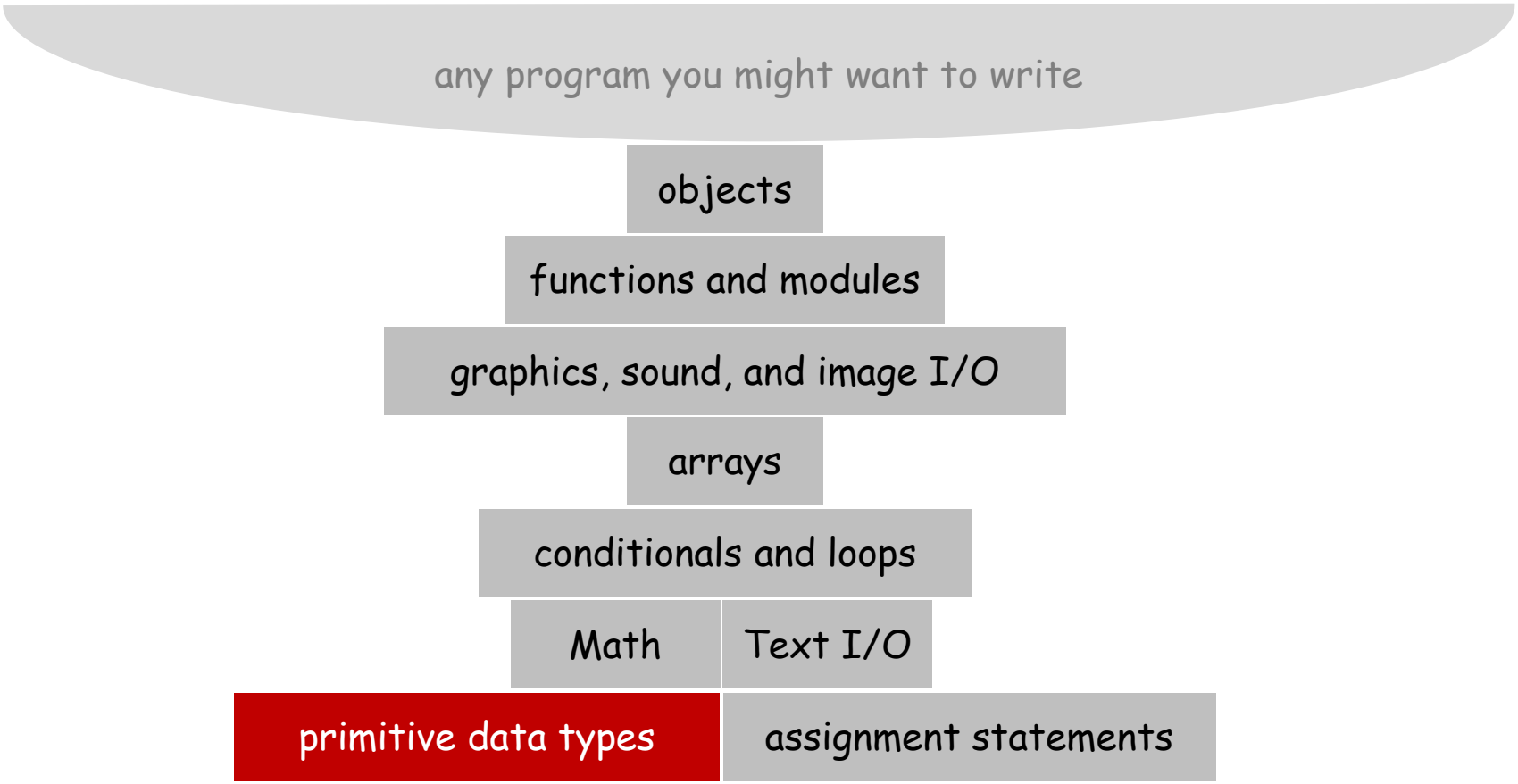
عناصر برنامه‌نویسی:
انواع داده‌ای پیش‌ساخته

سید ناصر رضوی **www.snrazavi.ir**
۱۳۹۵

۱-۲ انواع داده‌ای پیش‌ساخته



اجزای برنامه‌نویسی



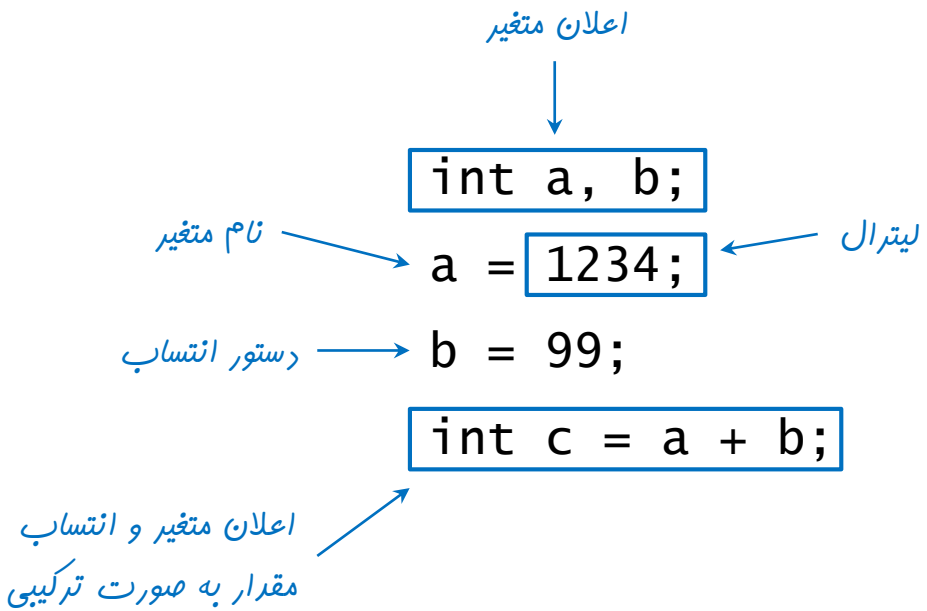
انواع داده‌ای پیش ساخته

- نوع داده‌ای. یک مجموعه از مقادیر به همراه عملیات تعریف شده بر روی آن مقادیر.

type	set of values	literal values	operations
`char`	characters	`'A'` `'@'`	compare
`String`	sequence of characters	`"Hello World"` `"Java is fun"`	+
`int`	integers	`17` `12345`	+, -, *, /
`double`	floating-point numbers	`3.1415` `6.022e23`	+, -, *, /
`boolean`	truth values	`true` `false`	&&, ||, !

تعاریف اولیه

- متغیر. نامی که به یک مقدار اشاره دارد.
- دستور انتساب. قرار دادن مقدار درون یک متغیر.



ردیابی

- ردیابی. یک جدول شامل مقادیر متغیرها پس از اجرای هر دستور.

	a	b	t
`int a, b;`	-	-	
`a = 1234;`	1234	-	
`b = 99;`	1234	99	
`int t = a;`	1234	99	1234
`a = b;`	99	99	1234
`b = t;`	99	1234	1234

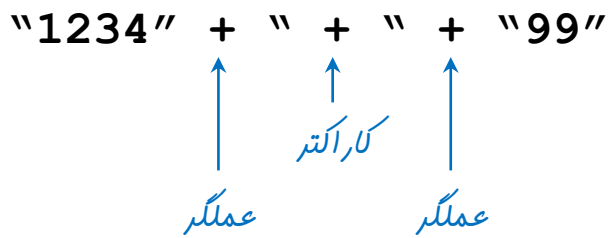
کار با داده‌های متنی

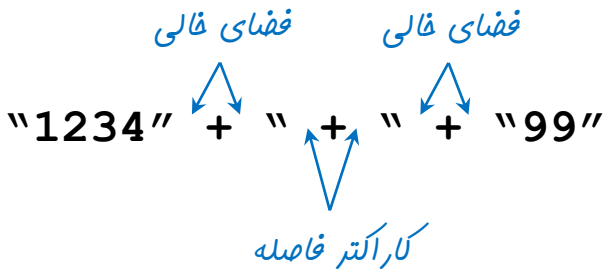
- نوع داده‌ای رشته. مفید برای عملیات ورودی و خروجی برنامه.

values	sequence of characters
typical literals	“Hello,” “1 “ “ * “
operation	concatenate
operator	+

expression	*value*
“Hi, “ + “Bob”	“Hi, Bob”
“1” + “ 2 “ + “1”	“1 2 1”
“1234” + “ + “ + “99”	“1234 + 99”
“1234” + “99”	“123499”

هشدار. معنای کاراکترها به محل استفاده‌ی آنها بستگی دارد.





مثال: بخش‌بندی‌های یک خط کش


```
public class Ruler {
   public static void main(String[] args) {
      String ruler1 = "1";
      String ruler2 = ruler1 + " 2 " + ruler1;
      String ruler3 = ruler2 + " 3 " + ruler2;
      String ruler4 = ruler3 + " 4 " + ruler3;
      System.out.println(ruler4);
   }
}
```

اتصال رشته‌ای

```
"1"
"1 2 1"
"1 2 1 3 1 2 1"
```

```
% java Ruler
1 2 1 3 1 2 1 4 1 2 1 3 1 2 1
```

اعداد صحیح

- نوع داده‌ای عدد صحیح. مفید برای بیان الگوریتم‌ها.

values	-2^{31} ... $2^{31} - 1$				
typical literals	1234	99	-99	0	1000000
operations	add	subtract	multiply	division	remainder
operators	+	-	*	/	%

اعداد صحیح

- نوع داده‌ای عدد صحیح. مفید برای بیان الگوریتم‌ها.

expression	*value*	*comment*
5 + 3	8	جمع
5 - 3	2	تفریق
5 * 3	15	ضرب
5 / 3	1	تقسیم (بدون قسمت اعشاری)
5 % 3	2	باقیمانده
1 / 0	-	خطای زمان اجرا

مثال: عملیات بر روی اعداد صحیح


```
public class IntOps {
   public static void main(String[] args) {
      int a = Integer.parseInt(args[0]);
      int b = Integer.parseInt(args[1]);
      int sum  = a + b;
      int prod = a * b;
      int quot = a / b;
      int rem  = a % b;
      System.out.println(a + " + " + b + " = " + sum );
      System.out.println(a + " * " + b + " = " + prod);
      System.out.println(a + " / " + b + " = " + quot);
      System.out.println(a + " % " + b + " = " + rem );
   }
}
```

آرگومان‌های خط فرمان

جاوا به صورت خودکار این مقادیر را به نوع رشته‌ای تبدیل می‌کند.

```
% javac IntOps.java
% java IntOps 1234 99
1234 + 99 = 1333
1234 * 99 = 122166
1234 / 99 = 12
1234 % 99 = 46
```

1234 = 12*99 + 46

اعداد اعشاری

- نوع داده‌ای **double**. مفید در محاسبات علمی.

values	real numbers (specified by IEEE 754 standard)
typical literals	3.14159 6.022e23 -3.0 2.0 1.4142135623730951
operations	add subtract multiply division
operators	+ - * /

اعداد اعشاری

- نوع داده‌ای **double**. مفید در محاسبات علمی.

expression	*value*
`3.141 + .03`	`3.171`
`3.141 - .03`	`3.111`
`6.02e23 / 2.0`	`3.01e23`
`5.0 / 3.0`	`1.6666666666666667`
`10.0 % 3.141`	`0.577`
`1.0 / 0.0`	`Infinity`
`Math.sqrt(2.0)`	`1.4142135623730951`
`Math.sqrt(-1.0)`	`NaN`

کتابخانه Math در جاوا

public class Math

این سه تابع برای انواع `int`، `long` و `float` نیز تعریف شده اند.

`double abs(double a)`	قدر مطلق مقدار a
`double max(double a, double b)`	ماکزیمم مقادیر a و b
`double min(double a, double b)`	مینیمم مقادیر a و b
`double sin(double theta)`	تابع سینوس
`double cos(double theta)`	تابع کسینوس
`double tan(double theta)`	تابع تانژانت
`double exp(double a)`	توان رسانی (e^a)
`double log(double a)`	لگاریتم طبیعی ($\ln a$)
`double pow(double a, double b)`	a به توان b
`long round(double a)`	گرد کردن به نزدیک‌ترین عدد صحیح
`double random()`	تولید یک عدد تصادفی در بازه $[0, 1)$
`double sqrt(double a)`	ریشه دوم a
`double E`	مقدار e (ثابت)
`double PI`	مقدار π (ثابت)

زاویه‌ها برحسب رادیان

http://java.sun.com/javase/6/docs/api/java/lang/Math.html

کتابخانه Math در جاوا

public class Math

double abs(double a)	قدر مطلق مقدار a
double max(double a, double b)	ماکزیمم مقادیر a و b
double min(double a, double b)	مینیمم مقادیر a و b
double sin(double theta)	تابع سینوس
double cos(double theta)	تابع کسینوس
double tan(double theta)	تابع تانژانت
double exp(double a)	توان رسانی (e^a)
double log(double a)	لگاریتم طبیعی (ln a)
double pow(double a, double b)	a به توان b
double random()	تولید یک عدد تصادفی در بازه [0, 1)
double sqrt(double a)	ریشه دوم a
double E	مقدار e (ثابت)
double PI	مقدار π (ثابت)

این سه تابع برای انواع int، long و float نیز تعریف شده‌اند.

زاویه‌ها برحسب رادیان

مثال: معادله‌ی درجه دوم

- مثال. حل معادله‌ی درجه دوم $x^2 + bx + c = 0$.

```
public class Quadratic {

   public static void main(String[] args) {
      // parse coefficients from command-line
      double b = Double.parseDouble(args[0]);
      double c = Double.parseDouble(args[1]);

      // calculate roots
      double discriminant = b * b - 4.0 * c;
      double d = Math.sqrt(discriminant);
      double root1 = (-b + d) / 2.0;
      double root2 = (-b - d) / 2.0;

      // print them out
      System.out.println(root1);
      System.out.println(root2);
   }
}
```

$$roots = \frac{-b \pm \sqrt{b^2 - 4c}}{2}$$

آزمایش

- آزمایش. بررسی خروجی به ازای ورودی‌های معتبر و نامعتبر.

```
% java Quadratic -3.0 2.0
2.0
1.0

% java Quadratic -1.0 -1.0
1.618033988749895
-0.618033988749895

% java Quadratic 1.0 1.0
NaN
NaN

% java Quadratic 1.0 hello
java.lang.NumberFormatException: hello

% java Quadratic 1.0
java.lang.ArrayIndexOutOfBoundsException
```

آرگومان‌های خط فرمان

نسبت طلایی

Not a Number

$x^2 - 3x + 2 = 0$

$x^2 - x - 1 = 0$

$x^2 + x + 1 = 0$

مقادیر بولی

- نوع داده‌ای boolean. مفید برای کنترل منطق و روند اجرای برنامه.

values	true or false		
literals	`true` `false`		
operations	and	or	not
operators	&&	||	!

a	!a
true	false
false	true

a	b	a && b	a || b
false	false	false	false
false	true	false	true
true	false	false	true
true	true	true	true

Truth-table definitions of boolean *operations*

عملگرهای مقایسه‌ای

- عملگرهای مقایسه‌ای. عملوندهایی از یک نوع را دریافت و یک عملوند از نوع بولی تولید می‌کنند.

op	*meaning*	`true`	`false`
`==`	*equal*	`2 == 2`	`2 == 3`
`!=`	*not equal*	`3 != 2`	`2 != 2`
`<`	*less than*	`2 < 13`	`2 < 2`
`<=`	*less than or equal*	`2 <= 2`	`3 <= 2`
`>`	*greater than*	`13 > 2`	`2 > 13`
`>=`	*greater than or equal*	`3 >= 2`	`2 >= 3`

Comparisons with `int` *operands and a* `boolean` *result*

مثال: سال کبیسه

- پرسش. آیا یک سال داده شده یک سال کبیسه است؟
- پاسخ. بله اگر (۱) بر ۴۰۰ بخش‌پذیر باشد یا (۲) بر ۴ بخش‌پذیر باشد اما بر ۱۰۰ نه.

```
public class LeapYear {

   public static void main(String[] args) {

      int year = Integer.parseInt(args[0]);
      boolean isLeapYear;

      // divisible by 4 but not 100
      isLeapYear = (year % 4 == 0) && (year % 100 != 0);

      // or divisible by 400
      isLeapYear = isLeapYear || (year % 400 == 0);
      System.out.println(isLeapYear);
   }
}
```

```
% java LeapYear 2004
true

% java LeapYear 1900
false

% java LeapYear 2000
true
```

تبدیل نوع

- تبدیل نوع. تبدیل از یک نوع به نوع دیگر.
  - تبدیل خودکار: بدون از دست رفتن دقت؛ یا تبدیل به رشته
  - صریح: با استفاده تبدیل نوع یا متدها

expression	*expression type*	*expression value*
`"1234" + 99`	`String`	`"123499"`
`Integer.parseInt("123")`	`int`	`123`
`(int) 2.71828`	`int`	`2`
`Math.round(2.71828)`	`long`	`3`
`(int) Math.round(2.71828)`	`int`	`3`
`(int) Math.round(3.14159)`	`int`	`3`
`11 * 0.3`	`double`	`3.3`
`(int) 11 * 0.3`	`double`	`3.3`
`11 * (int) 0.3`	`int`	`0`
`(int) (11 * 0.3)`	`int`	`3`

Typical type conversions

تولید اعداد صحیح تصادفی

- مثال. تولید یک عدد تصادفی بین 0 و N - 1.

```
public class RandomInt {
   public static void main(String[] args) {
      int N = Integer.parseInt(args[0]);
      double r = Math.random();
      int n = (int) (r * N);

      System.out.println("random integer is " + n);
   }
}
```

رشته به عدد صحیح (متر)

اعشاری به عدد صحیح (صریح)

عدد صحیح به اعشاری (خودکار)

عدد صحیح به رشته (خودکار)

```
% java RandomInt 6
random integer is 3

% java RandomInt 6
random integer is 0

% java RandomInt 10000
random integer is 3184
```

خلاصه

- یک نوع داده‌ای بیانگر مجموعه‌ای از مقادیر به همراه عملیات قابل انجام بر روی آن مقادیر است.
  - رشته — پردازش متن
  - اعداد صحیح و اعشاری — محاسبات ریاضی
  - بولی — تصمیم‌گیری

یک مثال از تبدیل نوع نادرست

- در جاوا باید:
  - نوع مقادیر را تعریف کنید.
  - هر زمان لازم بود انواع داده‌ای را به یکدیگر تبدیل کنید.
- چرا باید نوع هر مقدار را مشخص کنیم؟
  - مثال: موشک آریان ۵ در سال ۱۹۹۶ به دلیل تبدیل نوع نادرست بلافاصله پس از برخاستن منفجر شد!
  - کامپایلر می‌تواند در انجام این کار به ما کمک کند.
  - در برخی سطوح باید تبدیل نوع انجام شود.