

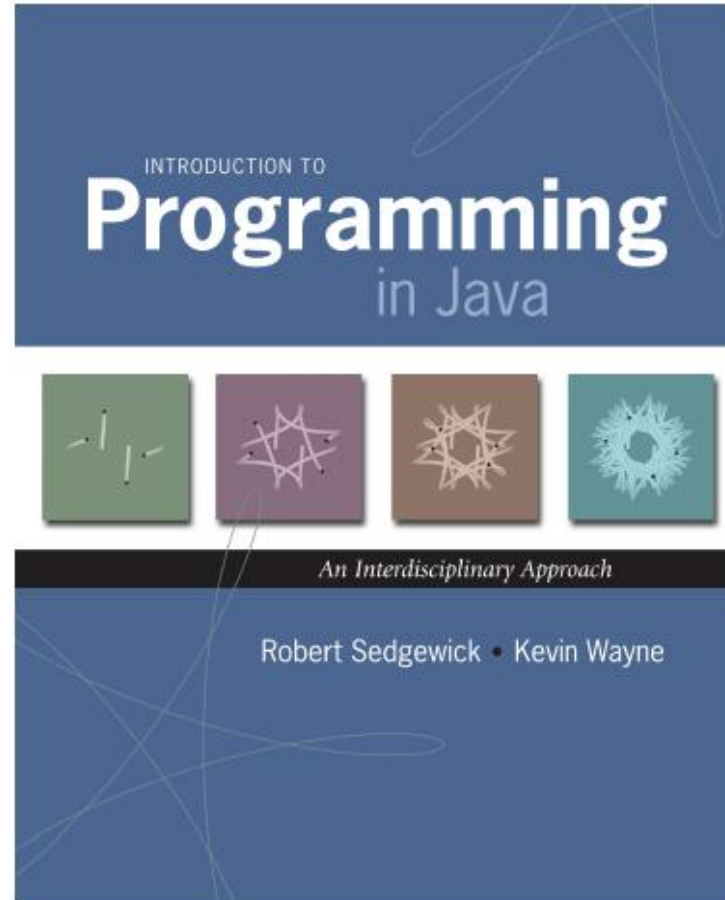
برنامه‌نویسی شی‌گرا: استفاده از انواع داده‌ای

سید ناصر رضوی www.snrazavi.ir

۱۳۹۶

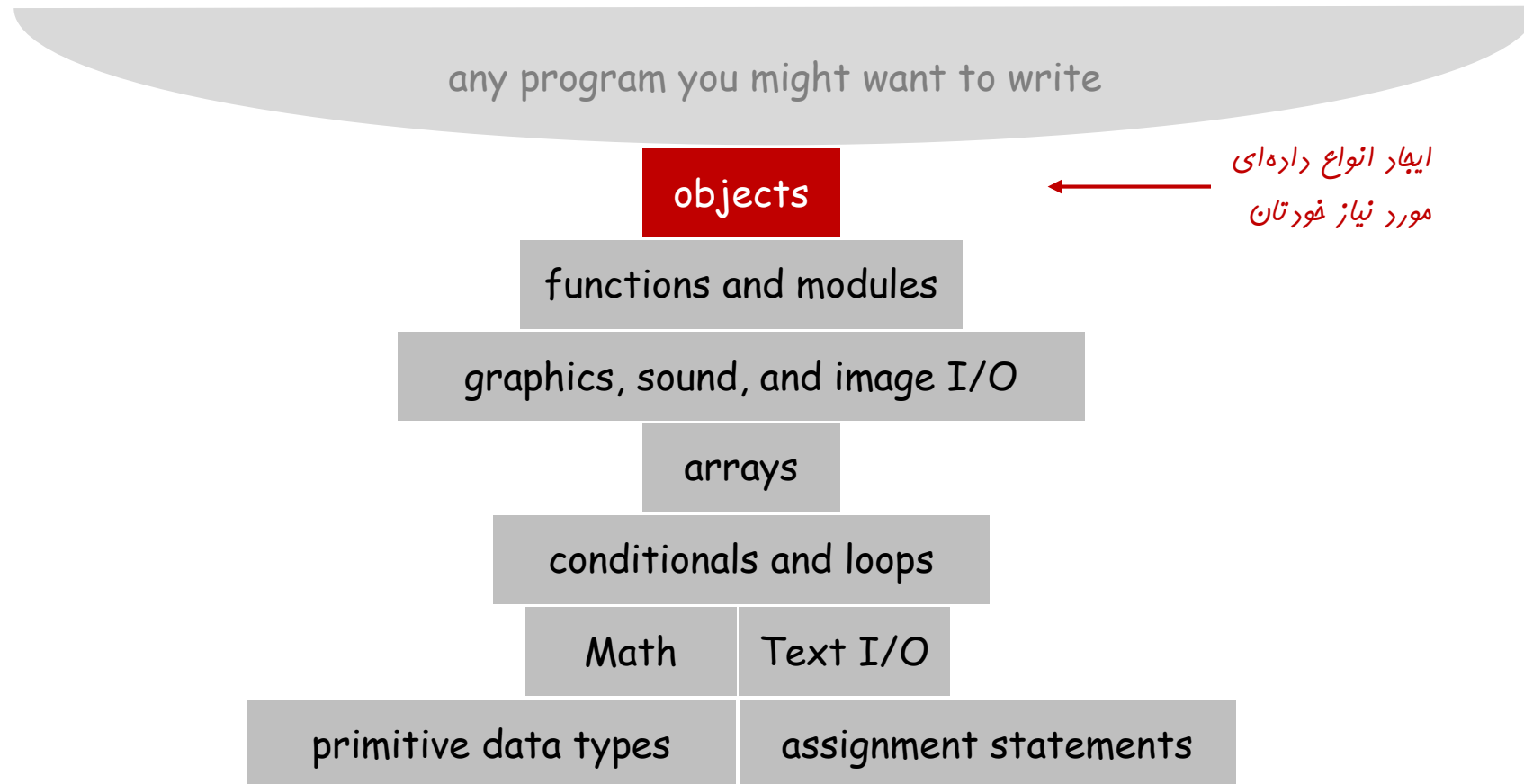
۳-۱ استفاده از انواع داده‌ای

۲



اجزای برنامه نویسی

۳



انواع داده‌ای

- نوع داده‌ای. یک مجموعه از مقادیر و عملیات قابل انجام بر روی آنها.
- انواع اولیه. عملیات مستقیماً به دستورالعمل‌های ماشین ترجمه می‌شوند.

Data Type	Values	Operations
boolean	true, false	not, and, or, xor
int	-2^{31} to $2^{31} - 1$	add, subtract, multiply
double	2^{64} possible real values	add, subtract, multiply

- می‌خواهیم برنامه‌هایی بنویسیم که بتوانند هر نوع از داده‌ها را پردازش کنند.
 - رنگ‌ها، تصاویر، رشته‌ها، جریان‌های ورودی، ...
 - اعداد مختلط، بردارها، ماتریس‌ها، چندجمله‌ای‌ها، ...
 - نقطه‌ها، چندضلعی‌ها، ذرات باردار، اجرام آسمانی، ...

□ شی. دربردارنده مقدار یک نوع داده‌ای؛ نام متغیر به شی ارجاع می‌دهد.

□ کاربرد.

□ به وسیله اشیا می‌توانیم هر نوع داده‌ای را که نیاز داریم، خودمان ایجاد کنیم.

□ سپس عملیات لازم را بر روی آنها تعریف و از آنها در برنامه‌هایمان استفاده می‌کنیم.

Data Type	Values	Operations
Color	24 bits (R, G, B)	get red component, brighten
Picture	2D array of colors	get or set color of pixel (i, j)
String	sequence of characters	length, substring, compare

سازنده‌ها و متدها

۶

- برای ایجاد یک شی جدید: از کلمه کلیدی **new** و نام نوع داده‌ای استفاده کنید.
- برای انجام یک عمل بر روی یک شی: از نام شی، عمگر نقطه، و نام متد مورد نظر استفاده کنید.

اعلان یک متغیر (نام شی)

فراخوانی سازنده برای ایجاد یک شی

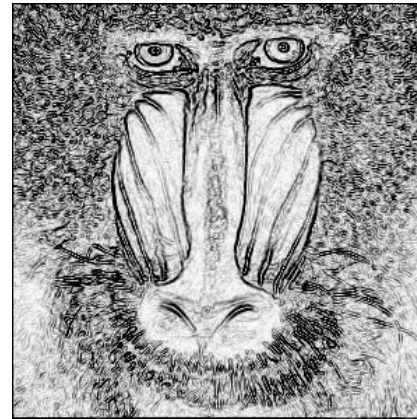
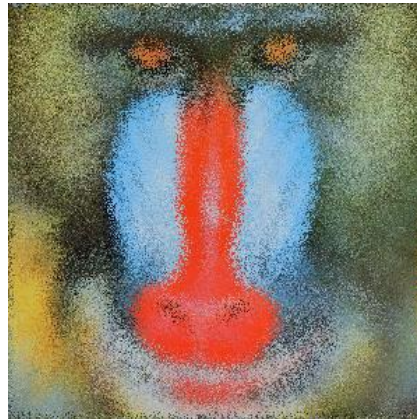
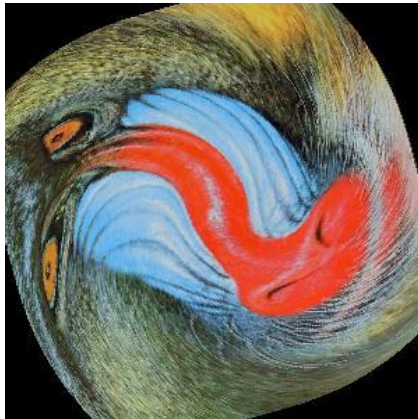
```
String s;  
s = new String("Hello, world");  
System.out.println(s.substring(0, 5));
```

نام شی

فراخوانی یک متد به منظور انجام عملیات بر روی مقدار شی

پردازش تصویر








۷



نوع داده‌ای رنگ

□ رنگ. درک چشم از تشعشعات الکترومغناطیس.

□ مجموعه مقادیر. [نمایش RGB] 256^3 مقدار ممکن، که بیانگر مقدار قرمز، سبز و آبی هر یک در بازه صفر و ۲۵۵ هستند.

R	G	B	Color
255	0	0	
0	255	0	
0	0	255	
0	0	0	
255	255	255	
255	0	255	
105	105	105	

نوع داده‌ای رنگ

۹

□ رنگ. درک چشم از تشعشعات الکترومغناطیس.

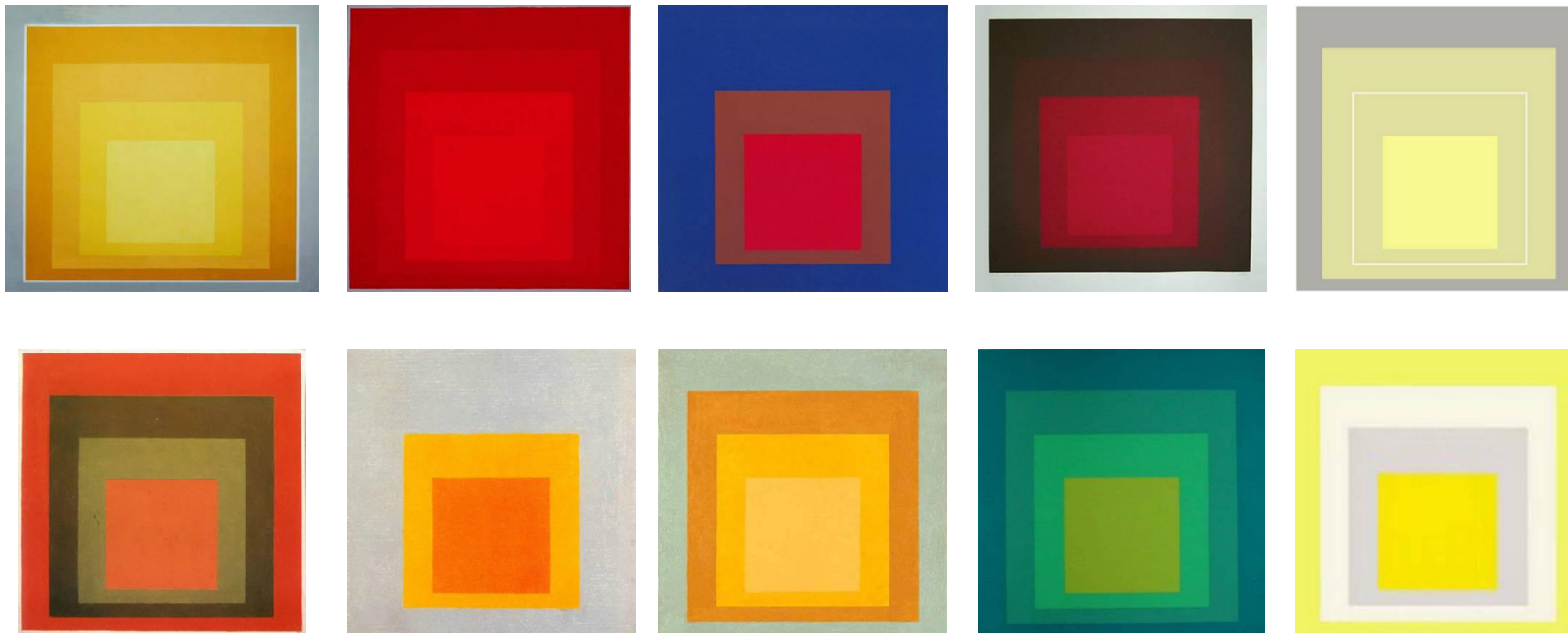
□ مجموعه مقادیر. [نمایش RGB] 256^3 مقدار ممکن، که بیانگر مقدار قرمز، سبز و آبی هر یک در بازه صفر و ۲۵۵ هستند.

```
public class java.awt.Color
```

<code>Color(int r, int g, int b)</code>	
<code>int getRed()</code>	شدت قرمز
<code>int getGreen()</code>	شدت سبز
<code>int getBlue()</code>	شدت آبی
<code>Color brighter()</code>	یک نسخه روشن‌تر از این رنگ
<code>Color darker()</code>	یک نسخه تیره‌تر از این رنگ
<code>String toString()</code>	نمایش رشته‌ای این رنگ
<code>boolean equals(Color c)</code>	آیا این رنگ با رنگ C برابر است؟

مربع‌های آلبرز

□ ژوزف آلبرز. شیوه نگرش مردم به رنگ‌ها را دگرگون نمود.



مربع‌های ژوزف آلبرز (۱۹۴۹ - ۱۹۷۵)

مربع‌های آلبرز

□ ژوزف آلبرز. شیوه نگرش مردم به رنگ‌ها را دگرگون نمود.

آبی خاکستری
↓ ↓

```
% java AlbersSquares 9 90 160 100 100 100
```



استفاده از رنگ‌ها در جاوا

۱۲

```
import java.awt.Color;

public class AlbersSquares {
    public static void main(String[] args) {
        int r1 = Integer.parseInt(args[0]);
        int g1 = Integer.parseInt(args[1]);
        int b1 = Integer.parseInt(args[2]);
        Color c1 = new Color(r1, g1, b1);

        int r2 = Integer.parseInt(args[3]);
        int g2 = Integer.parseInt(args[4]);
        int b2 = Integer.parseInt(args[5]);
        Color c2 = new Color(r2, g2, b2);

        StdDraw.setPenColor(c1);
        StdDraw.filledSquare(.25, .5, .2);
        StdDraw.setPenColor(c2);
        StdDraw.filledSquare(.25, .5, .1);

        StdDraw.setPenColor(c2);
        StdDraw.filledSquare(.75, .5, .2);
        StdDraw.setPenColor(c1);
        StdDraw.filledSquare(.75, .5, .1);
    }
}
```

رنگ اول

رنگ دوم

مربع اول

مربع دوم

% java AlbersSquares 9 90 160 100 100 100



روشنایی تک رنگ

۱۳

- روشنایی تک رنگ. درخشندگی مؤثر یک رنگ.
- فرمول NTSC.

$$Y = 0.299r + 0.587g + 0.114b$$

```
import java.awt.Color;

public class Luminance {
    public static double lum(Color c) {
        int r = c.getRed();
        int g = c.getGreen();
        int b = c.getBlue();
        return .299*r + .587*g + .114*b;
    }
}
```

سازگاری رنگ

۱۴

□ س. بر روی صفحه نمایش کامپیوتر و تلفن‌های همراه، کدام رنگ قلم بر روی کدام رنگ پس‌زمینه خواناتر است؟

□ پ. یک قاعده سرانگشتی. اختلاف روشنایی باید بزرگ‌تر یا مساوی ۱۲۸ باشد.



```
public static boolean compatible(Color a, Color b)
{
    return Math.abs(lum(a) - lum(b)) >= 128.0;
}
```

سطح خاکستری




۱۵

□ سطح خاکستری. زمانی که مقدار مؤلفه‌های R، G و B با هم برابر باشند، رنگ حاصل یک رنگ خاکستری در بازه صفر (سیاه) تا ۲۵۵ (سفید) خواهد بود.

□ تبدیل به خاکستری. استفاده از روشی به عنوان مقدار مؤلفه‌ها.

```
public static Color toGray(Color c)
{
    int y = (int) Math.round(lum(c));
    Color gray = new Color(y, y, y);
    return gray;
}
```

گرد کردن یک عدد اعشاری به نزدیک‌ترین عدد صحیح

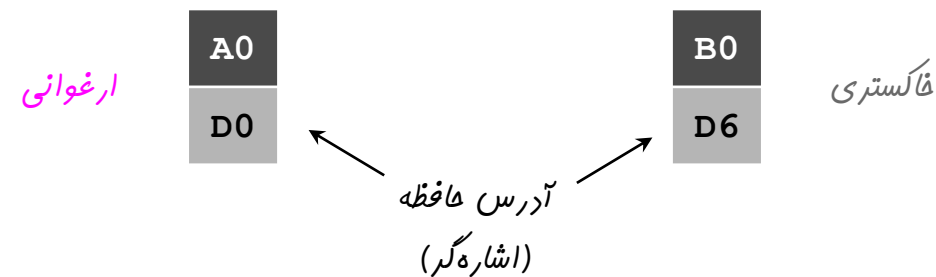
R	G	B		
9	90	166		رنگ
74	74	74		نصفه خاکستری
0	0	0		سیاه

$$0.299 * 9 + 0.587 * 90 + 0.114 * 166 = 74.445$$

شی‌گرایی

□ نمایش در حافظه.

D0	D1	D2	D3	D4	D5	D6	D7	D8
255	0	255	0	0	0	105	105	105



□ ارجاع به شی مشابه با نام متغیر است.

□ می‌توانیم مقدار ذخیره شده در آن را دستکاری کنیم.

□ می‌توان آن را به یک متد ارسال کنیم و یا به عنوان خروجی از متد دریافت کنیم.

ارجاع به اشیا

۱۷



□ رنه مارگریت. « این یک پیپ نیست. »

□ جاوا. این یک رنگ نیست.

```
Color sienna = new Color(160, 82, 45);  
Color c = sienna.darker();
```

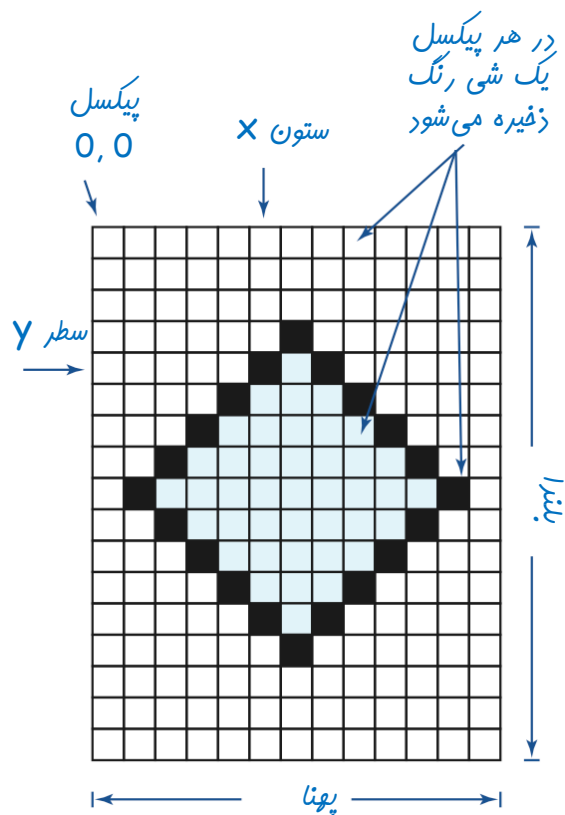
□ برنامه‌نویسی شی‌گرا. یک روش طبیعی به منظور مطالعه مدل‌های انتزاعی از دنیای واقعی.

نوع داده‌ای تصویر

۱۸

□ نوع داده‌ای تصویر. پایه پردازش تصویر.

□ مجموعه مقادیر. آرایه دو بعدی از اشیای رنگ (پیکسل‌ها).



```
public class Picture
```

```
    Picture(String filename)
```

ایجاد تصویر از یک فایل

```
    Picture(int w, int h)
```

ایجاد یک تصویر خالی با ابعاد w در h

```
    int width()
```

برگرداندن پهنای تصویر

```
    int height()
```

برگرداندن بلندای تصویر

```
    color get(int x, int y)
```

برگرداندن رنگ پیکسل (x, y)

```
    void set(int x, int y, color c)
```

تنظیم رنگ پیکسل (x, y) با c

```
    void show()
```

نمایش تصویر در یک پنجره

```
    void save(String filename)
```

ذخیره تصویر در یک فایل

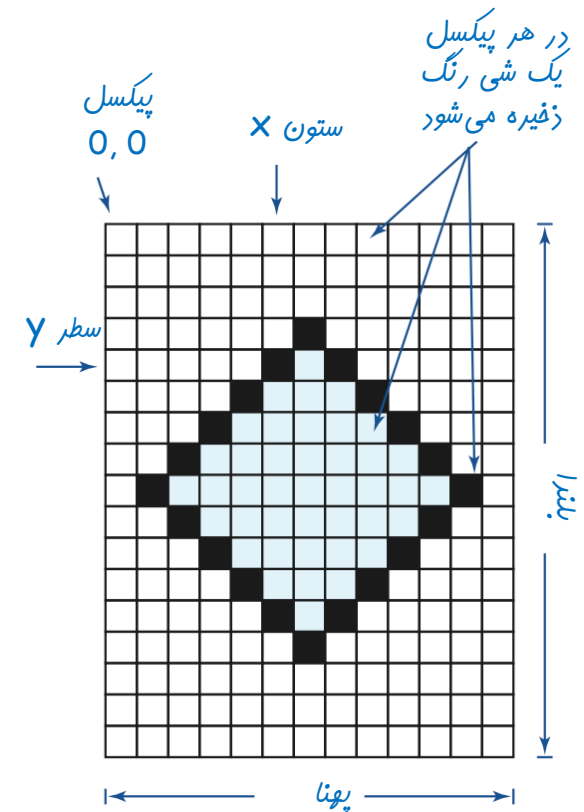
پردازش تصویر: فیلتر خاکستری

۱۹

□ هدف. تبدیل یک تصویر رنگی به یک تصویر خاکستری.

```
import java.awt.Color;

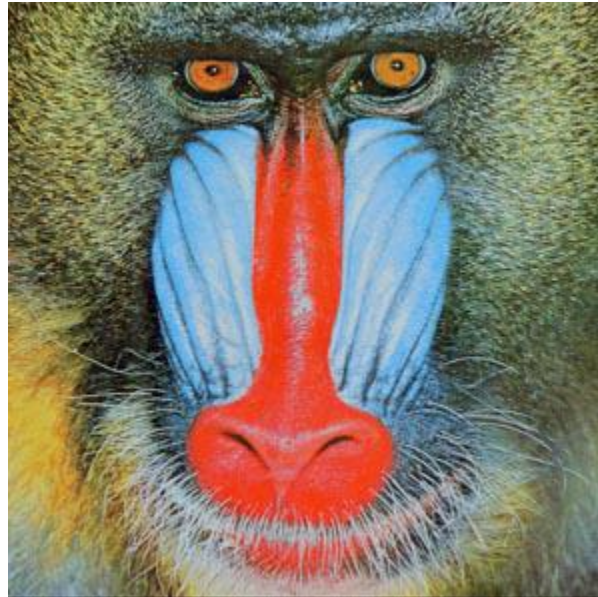
public class Grayscale {
    public static void main(String[] args) {
        Picture pic = new Picture(args[0]);
        for (int x = 0; x < pic.width(); x++) {
            for (int y = 0; y < pic.height(); y++) {
                Color color = pic.get(x, y);
                Color gray = Luminance.toGray(color);
                pic.set(x, y, gray);
            }
        }
        pic.show();
    }
}
```



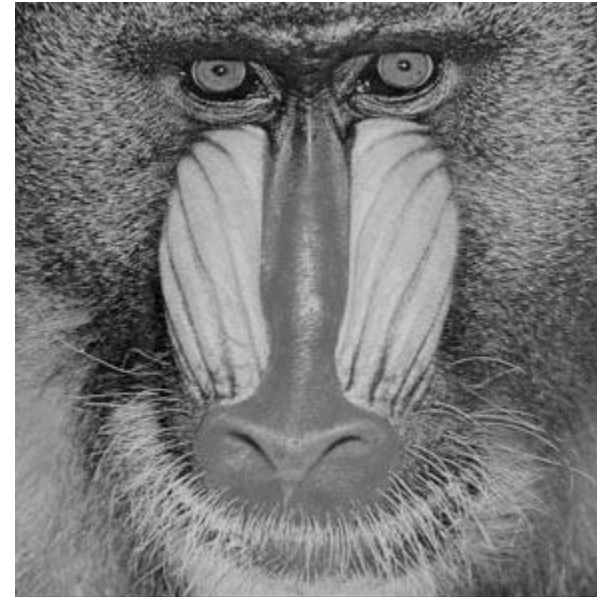
پردازش تصویر: فیلتر خاکستری

۲۰

□ هدف. تبدیل یک تصویر رنگی به یک تصویر خاکستری.



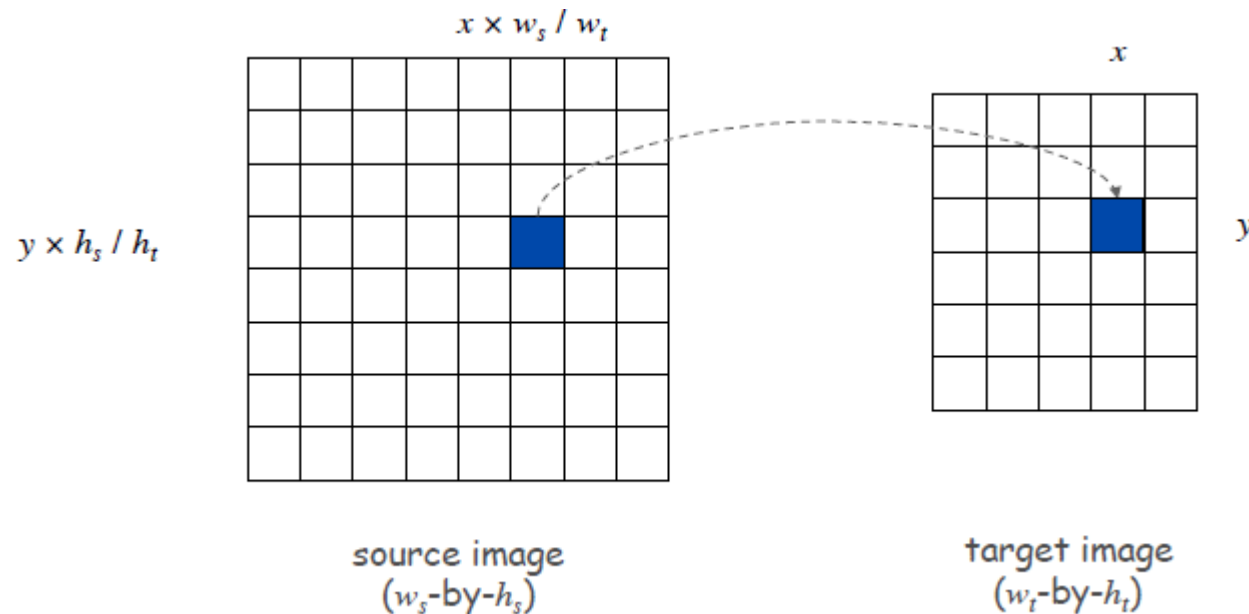
mandrill.jpg



% java Grayscale mandrill.jpg

پردازش تصویر: فیلتر تغییر اندازه

□ هدف. کوچک کردن یا بزرگ کردن اندازه یک تصویر.



پردازش تصویر: فیلتر تغییر اندازه

```
import java.awt.Color;

public class Scale {
    public static void main(String[] args) {
        String filename = args[0];
        int w = Integer.parseInt(args[1]);
        int h = Integer.parseInt(args[2]);

        Picture source = new Picture(filename);
        Picture target = new Picture(w, h);

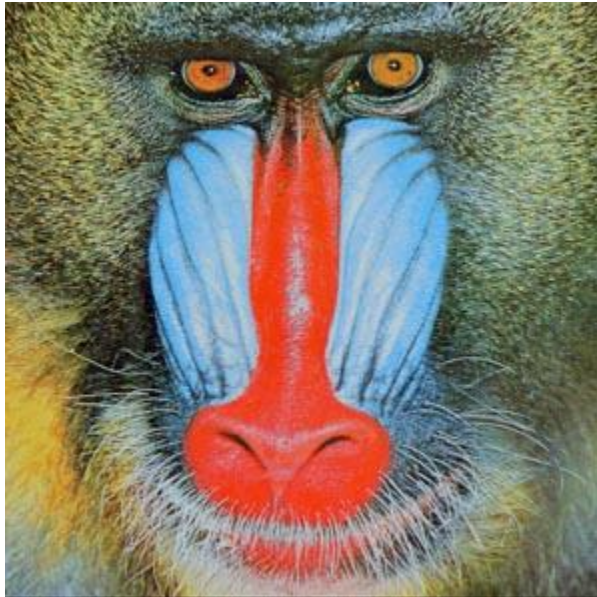
        for (int tx = 0; tx < target.width(); tx++) {
            for (int ty = 0; ty < target.height(); ty++) {
                int sx = tx * source.width() / target.width();
                int sy = ty * source.height() / target.height();
                Color color = source.get(sx, sy);
                target.set(tx, ty, color);
            }
        }

        source.show();
        target.show();
    }
}
```

پردازش تصویر: فیلتر تغییر اندازه

۲۳

□ فیلتر تغییر اندازه. ایجاد دو شی تصویر و دو پنجره.



mandrill.jpg (298-by-298)

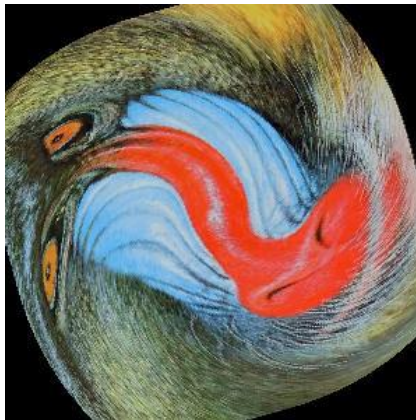


% java Scale mandrill.jpg 400 200

چند فیلتر پردازش تصویر دیگر



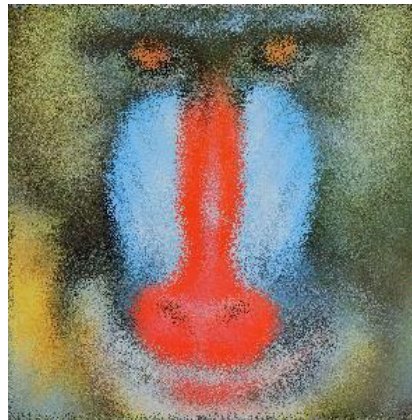
براسازی مؤلفه‌های RGB



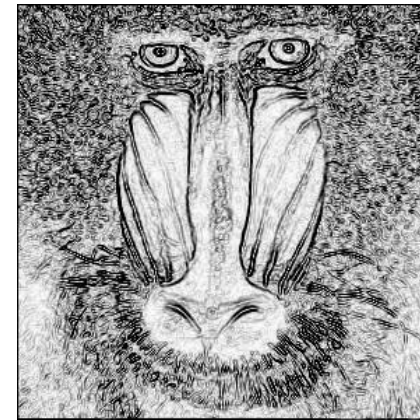
فیلتر گرداب



فیلتر موج



فیلتر شیشه



فیلتر لبه سوبل

فیلتر جداسازی رنگ‌ها

۲۵

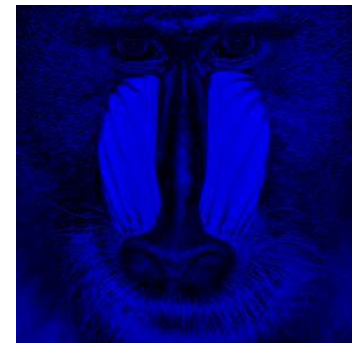
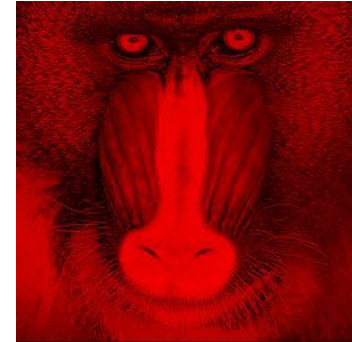
```
import java.awt.Color;

public class ColorSeparation {
    public static void main(String[] args) {
        Picture source = new Picture(args[0]);
        int w = source.width();
        int h = source.height();

        Picture R = new Picture(w, h);
        Picture G = new Picture(w, h);
        Picture B = new Picture(w, h);

        for (int x = 0; x < target.width(); x++) {
            for (int y = 0; y < target.height(); y++) {
                Color color = source.get(x, y);
                int r = color.getRed();
                int g = color.getGreen();
                int b = color.getBlue();
                R.set(x, y, new Color(r, 0, 0));
                G.set(x, y, new Color(0, g, 0));
                B.set(x, y, new Color(0, 0, b));
            }
        }

        R.show();
        G.show();
        B.show();
    }
}
```



نوع داده‌ای رشته

۲۶

□ نوع داده‌ای رشته. پایه پردازش متن.

□ مجموعه مقادیر. دنباله‌ای از کاراکترهای یونیکد.

```
public class String
```

```
    String(String s)
```

ایجاد یک رشته با مقدار *s*

```
    int length()
```

طول رشته

```
    char charAt(int i)
```

کاراکتر *i*ام

```
    String substring(int i, int j)
```

کاراکتر *i* تا کاراکتر *j* - 1

```
    boolean contains(String sub)
```

آیا رشته شامل زیررشته *sub* است؟

```
    boolean startsWith(String pre)
```

آیا رشته با *pre* شروع می‌شود؟

```
    boolean endsWith(String post)
```

آیا رشته با *post* پایان می‌یابد؟

```
    int indexOf(String p)
```

اندیس اولین رخداد *p*

```
    int indexOf(String p, int i)
```

اندیس اولین رخداد *p* از بعد از *i*

```
    String concat(String t)
```

اتصال این رشته با رشته *t*

```
    int compareTo(String t)
```

مقایسه دو رشته

چند مثال از پردازش رشته

<i>is the string a palindrome</i>	<pre>int N = s.length(); for (int i = 0; i < N/2; i++) if (s.charAt(i) != s.charAt(N - 1 - i)) return false; return true;</pre>
<i>extract file name and extension from a command-line argument</i>	<pre>String s = args[0]; int dot = s.indexOf("."); String base = s.substring(0, dot); String extension = s.substring(dot + 1, s.length());</pre>
<i>print all lines in standard that contains a string specified on the command-line</i>	<pre>String query = args[0]; while (!StdIn.isEmpty()) { String s = StdIn.readLine(); if (s.contains(query)) StdOut.println(s); }</pre>
<i>print all the hyperlinks in the text file on standard input</i>	<pre>while (!StdIn.isEmpty()) { String s = StdIn.readString(); if (s.startsWith("http://") && s.endsWith(".edu")) StdOut.println(s); }</pre>

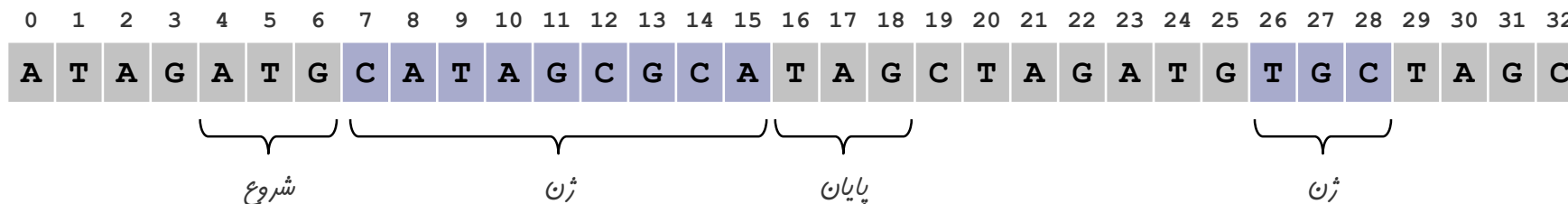
یافتن ژن

- ژنوم. نمایش ژنوم به عنوان یک رشته بر روی الفبای $\{A, C, T, G\}$.
- ژن. یک زیررشته از ژنوم که بیانگر یک واحد عملیاتی است.
 - با ATG شروع می‌شود. [کودون شروع]
 - مضربی از ۳ نوکلئوتید. [کودون‌های غیر از شروع و پایان]
 - با TAG، TAA یا TGA پایان می‌یابد. [کودون‌های پایان]
- هدف. یافتن تمام ژن‌ها.

0	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	30	31	32
A	T	A	G	A	T	G	C	A	T	A	G	C	G	C	A	T	A	G	C	T	A	G	A	T	G	T	G	C	T	A	G	C

یافتن ژن

- ژنوم. نمایش ژنوم به عنوان یک رشته بر روی الفبای $\{A, C, T, G\}$.
- ژن. یک زیررشته از ژنوم که بیانگر یک واحد عملیاتی است.
 - با ATG شروع می‌شود. [کودون شروع]
 - مضربی از ۳ نوکلئوتید. [کودون‌های غیر از شروع و پایان]
 - با TAG، TAA یا TGA پایان می‌یابد. [کودون‌های پایان]
- هدف. یافتن تمام ژن‌ها.



یافتن ژن: الگوریتم

□ الگوریتم. ژنوم را از چپ به راست پویش کن.

□ اگر کودون شروع، آنگاه مقدار **beg** را برابر با اندیس **i** قرار بده.

□ اگر کودون پایان و زیررشته مضربی از سه است:

■ ژن را به خروجی بفرست.

■ مقدار **beg** را برابر با **-1** قرار بده.

i	codon		beg	gene	input string
	start	stop			
0			-1		ATAGATGCATAGCGCATAGCTAGATGTGCTAGC
1		TAG	-1		ATAGATGCATAGCGCATAGCTAGATGTGCTAGC
4	ATG		4		ATAGATGCATAGCGCATAGCTAGATGTGCTAGC
9		TAG	4	$\underbrace{\hspace{2cm}}_{\text{مضرب ۳}}$	ATAGATGCATAGCGCATAGCTAGATGTGCTAGC
16		TAG	4	CATAGCGCA	ATAGATGCATAGCGCATAGCTAGATGTGCTAGC
20		TAG	-1		ATAGATGCATAGCGCATAGCTAGATGTGCTAGC
23			23		ATAGATGCATAGCGCATAGCTAGATGTGCTAGC
29		TAG	23	TGC	ATAGATGCATAGCGCATAGCTAGATGTGCTAGC

یافتن ژن: پیاده‌سازی

۳۱

```
public class GeneFind {
    public static void main(String[] args) {
        String start = args[0];
        String stop = args[1];
        String genome = StdIn.readAll();

        int beg = -1;
        for (int i = 0; i < genome.length() - 2; i++) {
            String codon = genome.substring(i, i+3);
            if (codon.equals(start)) beg = i;
            if (codon.equals(stop) && beg != -1) {
                String gene = genome.substring(beg+3, i);
                if (gene.length() % 3 == 0) {
                    StdOut.println(gene);
                    beg = -1;
                }
            }
        }
    }
}
```

```
% more genomeTiny.txt
ATAGATGCATAGCGCATAGCTAGATGTGCTAGC

% java GeneFind ATG TAG < genomeTiny.txt
CATAGCGCA
TGC
```

رشته‌ها در برنامه‌نویسی شی‌گرا

□ نمایش رشته در حافظه.

□ ژنوم = "aacaagtttacaagc"

D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	DA	DB	DC	DD	DE
a	a	c	a	a	g	t	t	t	a	c	a	a	g	c

genome

A0	A1
D0	15

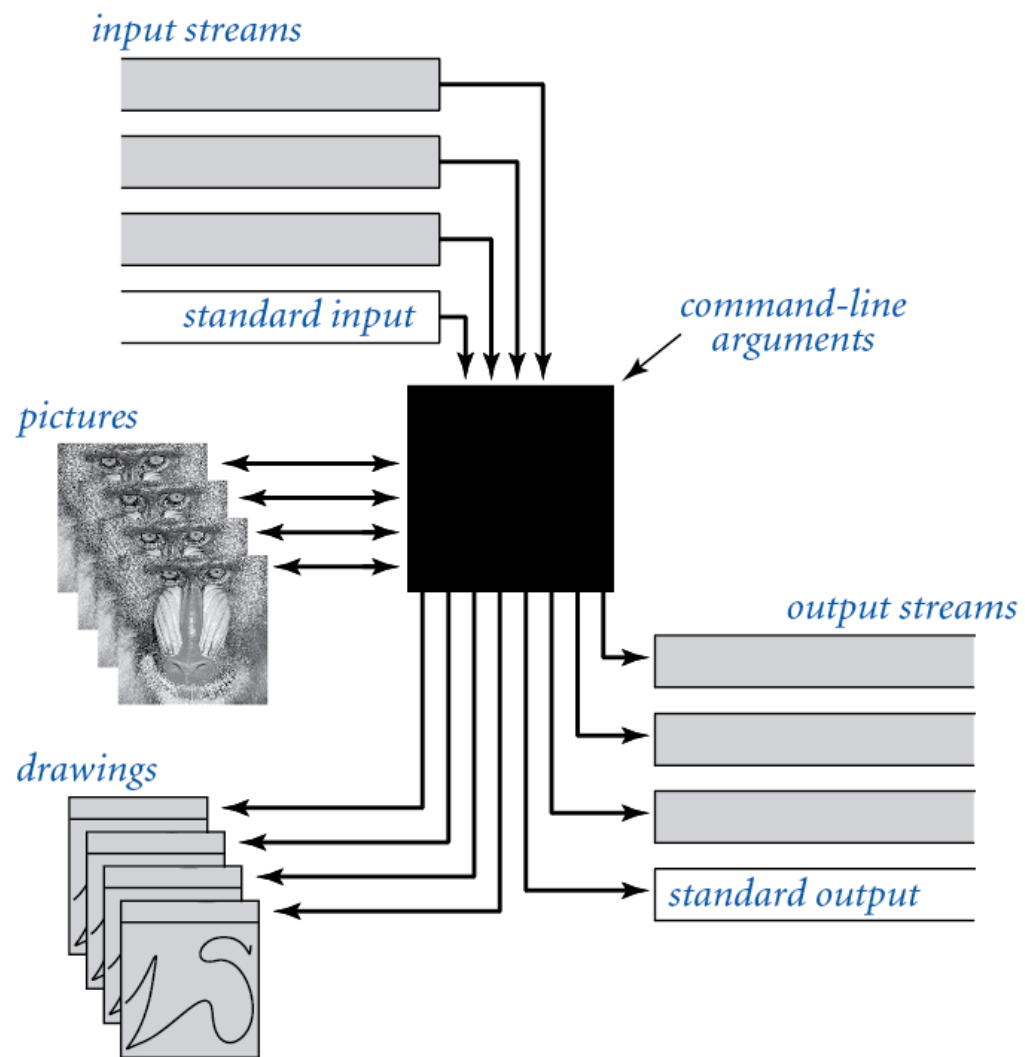
↑ آدرس حافظه
↑ طول رشته

- `s = genome.substring(1, 5)`
- `t = genome.substring(9, 13);`

s		t	
B0	B1	B2	B3
D1	4	D9	4

رشته‌های `s` و `t` دورشته متفاوت هستند که مقدار "aca" بین آنها مشترک است.

- `(s == t)` نادرست، مقایسه اشاره‌گرها
- `(s.equals(t))` درست، مقایسه دنباله کاراکترها



ورودی غیر استاندارد

۳۴

- ورودی استاندارد. خواندن از پنجره ترمینال.
- هدف. خواندن از چندین جریان ورودی مختلف.
- نوع داده‌ای **In**. خواندن ورودی‌های متنی از یک فایل، یک وب سایت یا شبکه.
- مثال. آیا دو فایل متنی با هم برابر هستند؟

```
public class Diff {  
    public static void main(String[] args) {  
        In in0    = new In(args[0]); ← فخواندن از یک فایل  
        In in1    = new In(args[1]); ← فخواندن از یک فایل دیگر  
        String s = in0.readAll();  
        String t = in1.readAll();  
        StdOut.println(s.equals(t));  
    }  
}
```

یافتن قیمت فعلی سهام گوگل

۳۵

GOOG: Summary for GOOGLE - Yahoo! Finance

http://finance.yahoo.com/q?s=goog

Google Movies Weather Tech News Sports CS Princeton Java 1.5 Book 1 Book 2 Courses Blogs

Yahoo! My Yahoo! Mail More Make Y! your home page New User? Sign Up Sign In Help

Dow ↓ 0.66% Nasdaq ↓ 0.55%

Search: Monday, March 26, 2007, 12:00PM ET - U.S. Markets close in 4 hours.

HOME INVESTING NEWS & OPINION PERSONAL FINANCE MY PORTFOLIOS

Enter Symbol(s) GET QUOTES Symbol Lookup Finance Search

Google Inc. (GOOG)

At 11:45AM ET: **459.52** ↓ 2.31 (0.50%)

MORE ON GOOG

- Quotes
 - Summary
 - Options
 - Historical Prices
- Charts
 - Basic Chart
 - Technical Analysis
- News & Info
 - Headlines
 - Financial Blogs
 - Company Events
 - Message Board
- Company

GOOGLE (NasdaqGS:GOOG) Delayed quote data

Last Trade:	459.52	Day's Range:	455.62 - 464.00
Trade Time:	11:45AM ET	52wk Range:	360.57 - 513.00
Change:	↓ 2.31 (0.50%)	Volume:	2,262,942
Prev Close:	461.83	Avg Vol (3m):	5,803,120
Open:	460.40	Market Cap:	142.89B
Bid:	459.61 x 200	P/E (ttm):	46.22
Ask:	459.64 x 100	EPS (ttm):	9.94
1y Target Est:	565.64	Div & Yield:	N/A (N/A)

New! Try our new Charts in Beta
GOOG 26-Mar 11:42am (C)Yahoo!

1d 5d 3m 6m 1y 2y 5y

Switch to TD AMERITRADE and trade free for 45 days.

NEW Add Quotes to Your Web Site Add GOOG to Portfolio Set Alert Download Data

We can no longer offer real-time ECN data due to pricing changes by the exchanges. For real-time quotes (including real-time pre/post market data), sign up for a free trial of [Real-time Quotes](#)

یافتن قیمت فعلی سهام گوگل

۳۶

```
...
<tr>
<td class="yfnc_tablehead1" width="48%">
Last Trade:
</td>
<td class="yfnc_tabledata1">
<big>
<b>459.52</b>
</big>
</td>
</tr>
<tr>
<td class="yfnc_tablehead1" width="48%">
Trade Time:
</td>
<td class="yfnc_tabledata1">
11:45AM ET
</td>
</tr>
...
```

یافتن قیمت فعلی سهام گوگل

۳۷

□ هدف. یافتن قیمت فعلی سهام گوگل.

□ خواندن فایل HTML از آدرس `http://finance.yahoo.com/q?s=`

□ یافتن اولین رشته محصور میان `` و `` پس از رشته `Last Trade`

```
public class StockQuote {
    public static void main(String[] args) {
        String name = "http://finance.yahoo.com/q?s=";
        In in = new In(name + args[0]);
        String input = in.readAll();
        int start = input.indexOf("Last Trade:", 0);
        int from = input.indexOf("<b>", start);
        int to = input.indexOf("</b>", from);
        String price = input.substring(from + 3, to);
        StdOut.println(price);
    }
}
```

```
% java StockQuote goog
616.50
```

خرید و فروش سهام



□ اندکی رنگ و لعاب.

- نمودار قیمت را به صورت بلادرنگ ترسیم کنید.
- اگر قیمت از یک حد مشخص پایین تر رفت، به کاربر خبر دهید.
- کد خود را برای تعیین زمان مناسب خرید و فروش تعیین کنید.
- به صورت خودکار به شرکت مورد نظر سفارش‌های خرید و فروش ارسال کنید.

□ **هشدار!** لطفاً، لطفاً، با مسئولیت خود و پذیرفتن ریسک مالی اقدام به انجام این کار نمایید!

برنامه‌نویسی شی‌گرا: خلاصه

- شی. دربرگیرنده مقدار یک نوع داده‌ای؛ نام متغیر به شی ارجاع می‌دهد.
- در جاوا برنامه‌ها بر روی اشاره‌گرها به اشیا عملیات انجام می‌دهند.
 - استثنا: انواع داده‌ای اولیه مانند اعداد صحیح، اعداد اعشاری و نوع بولی.
 - اشاره‌گرها: رشته، رنگ، تصویر، آرایه‌ها و هر چیز دیگر.
- سخن آخر. ما توانستیم برنامه‌هایی به منظور کار بر روی رنگ‌ها، تصاویر و رشته‌ها بنویسیم.
اولین گام در تسلط بر نامه‌نویسی شی‌گرا: کار با اشیای موجود
- جلسه آینده. نوشتن برنامه‌هایی به منظور کار بر روی اشیای تعریف شده به وسیله خود ما.
گام بعدی: تعریف اشیای جدید!