

# Chapter 1: JShell: A Read-Evaluate-Print-Loop for Java 9

```
bin — java · jshell — 78×24
Gastons-MacBook-Pro:~ gaston$ cd $(/usr/libexec/java_home)/bin
Gastons-MacBook-Pro:bin gaston$ pwd
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/bin
Gastons-MacBook-Pro:bin gaston$ javac -version
javac 9-ea
Gastons-MacBook-Pro:bin gaston$ ./jshell
| Welcome to JShell -- Version 9-ea
| For an introduction type: /help intro

jshell> █
```

```
Command Prompt - jshell
C:\Users\gaston>jshell
| Welcome to JShell -- Version 9-ea
| For an introduction type: /help intro

jshell> System.out.printf("Object-Oriented Programming rocks with Java 9!\n")
Object-Oriented Programming rocks with Java 9!
$1 ==> java.io.PrintStream@4c70fda8

jshell> System.out.printf("Object-Oriented Programming rocks with Java 9!\n");
Object-Oriented Programming rocks with Java 9!
$2 ==> java.io.PrintStream@4c70fda8

jshell>
```

```
Command Prompt - jshell
jshell> S
SafeVarargs Scanner
ScheduledExecutorService ScheduledFuture
ScheduledThreadPoolExecutor SecureCacheResponse
SecureDirectoryStream SecurityException
SecurityManager Semaphore
SequenceInputStream Serializable
SerializablePermission ServerSocket
ServiceConfigurationError ServiceLoader
Set Short
SimpleFileVisitor SimpleTimeZone
Socket SocketAddress
SocketException SocketImpl
SocketImplFactory SocketOption
SocketOptions SocketPermission
SocketTimeoutException SortedMap
SortedSet Spliterator
Spliterators SplittableRandom
Stack StackFramePermission
StackOverflowError StackTraceElement
StackWalker StandardCopyOption
StandardOpenOption StandardProtocolFamily
StandardSocketOptions StandardWatchEventKinds
Stream StreamCorruptedException
StreamSupport StreamTokenizer
StrictMath String
StringBuffer StringBufferInputStream
StringBuilder StringIndexOutOfBoundsException
StringJoiner StringReader
StringTokenizer StringWriter
SubmissionPublisher Supplier
SuppressWarnings SyncFailedException
SynchronousQueue System

jshell> S
```

```
Command Prompt - jshell
StringJoiner
StringTokenizer
SubmissionPublisher
SuppressWarnings
SynchronousQueue
StringReader
StringWriter
Supplier
SyncFailedException
System

jshell> System.out.
append(      checkError()  close()      equals(
flush()      format(      getClass()  hashCode()
notify()     notifyAll()  print(      printf(
println(     toString()   wait(       write(

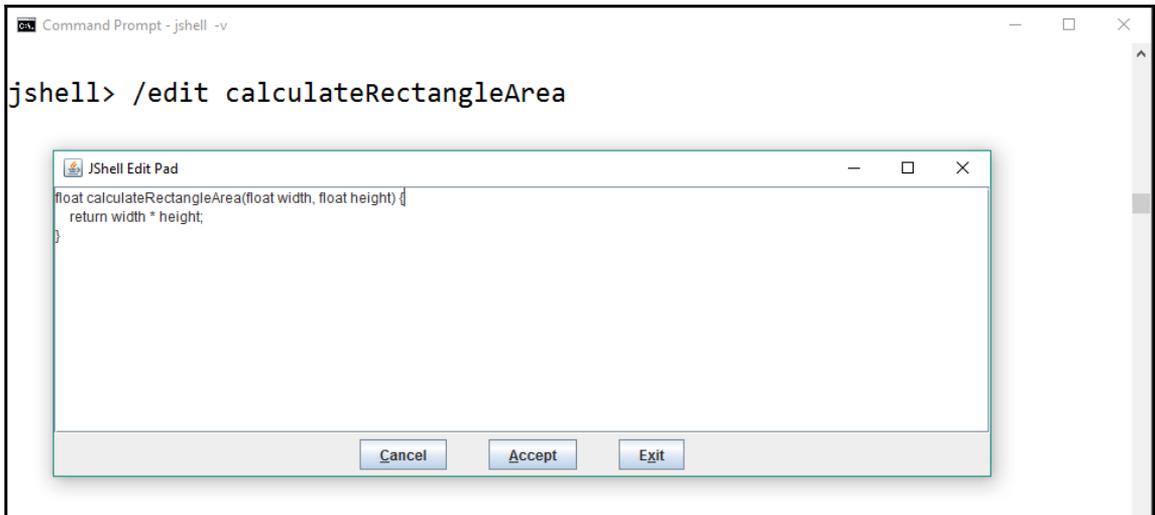
jshell> System.out.
```

```
Command Prompt - jshell
notify()     notifyAll()  print(      printf(
println(     toString()   wait(       write(

jshell> System.out.println(
println(

jshell> System.out.println("Auto-complete is helpful in JShell");
Auto-complete is helpful in JShell

jshell>
```



```
Command Prompt - jshell -v
| The '/set' command requires a sub-command. See: '/help /set'

jshell> /set editor "C:\\Program Files\\Sublime Text 3\\sublime_text.exe"
| Editor set to: C:\\Program Files\\Sublime Text 3\\sublime_text.exe

jshell> /edit calculateRectangleArea
```

C:\Users\gaston\AppData\Local\Temp\jshelltemp3750194570671088214\4872338378820725526.edit - Sublime Text (UNREGISTERED)

File Edit Selection Find View Goto Tools Project Preferences Help

4872338378820725526.edit x

```
1 float calculateRectangleArea(float width, float height) {
2     float area = width * height;
3     System.out.printf("Width: %.2f\n", width);
4     System.out.printf("Height: %.2f\n", height);
5     System.out.printf("Area: %.2f\n", area);
6     return area;
7 }
8
```

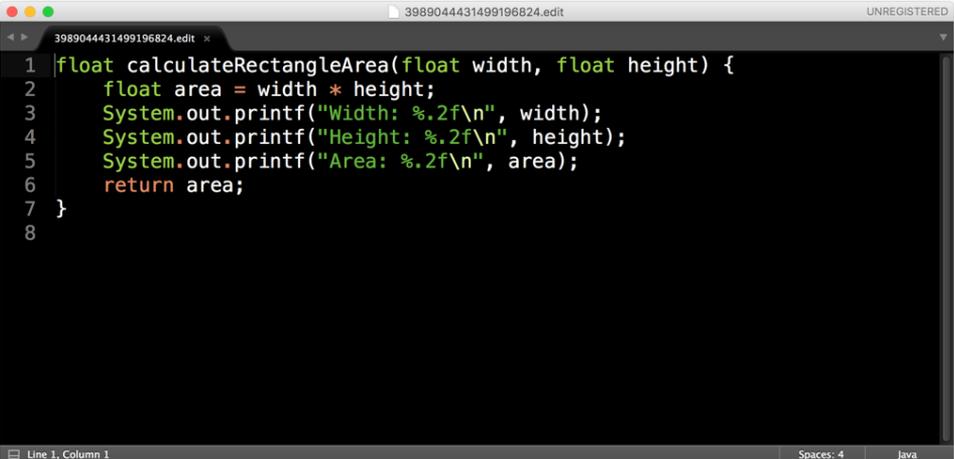
Line 3, Column 15      Tab Size: 4      Java

```
bin — java • jshell — 78x24

jshell> /set editor "/Applications/Sublime Text.app/Contents/SharedSupport/bin/
/subl"
| Editor set to: /Applications/Sublime Text.app/Contents/SharedSupport/bin/su
bl

jshell> /edit calculateRectangleArea

jshell> 
```

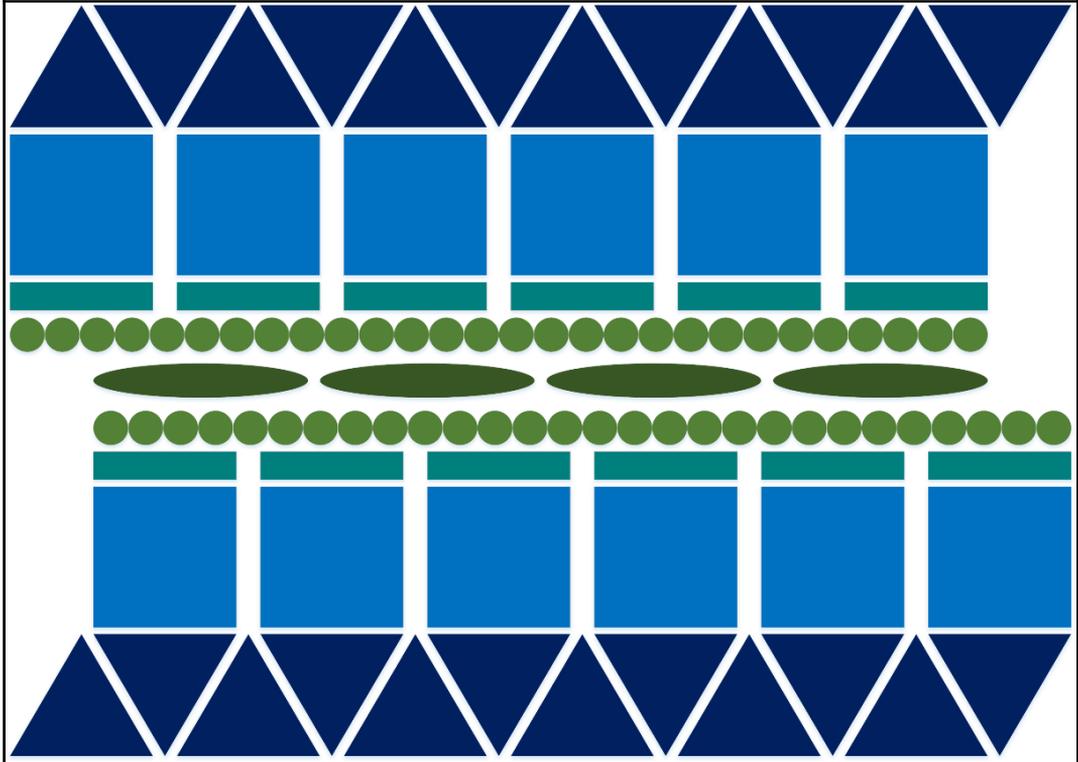


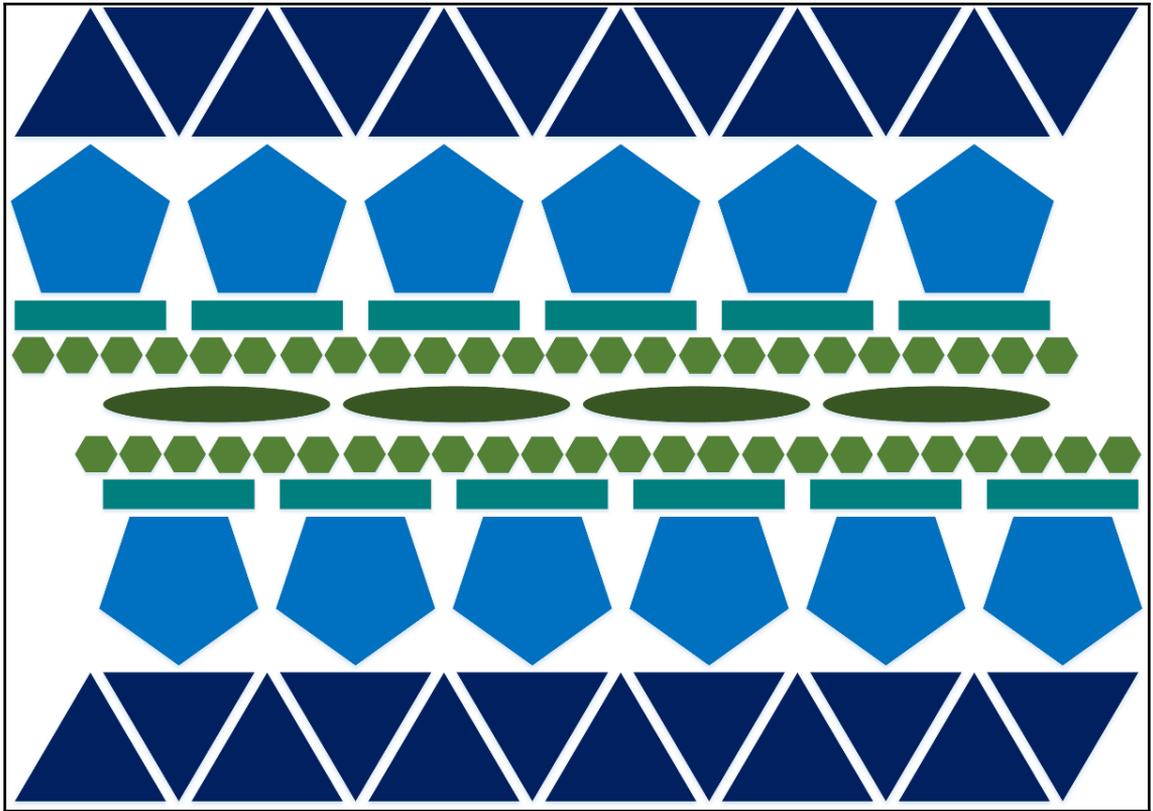
The screenshot shows a Java IDE window titled "3989044431499196824.edit" with a dark theme. The code editor contains the following Java code:

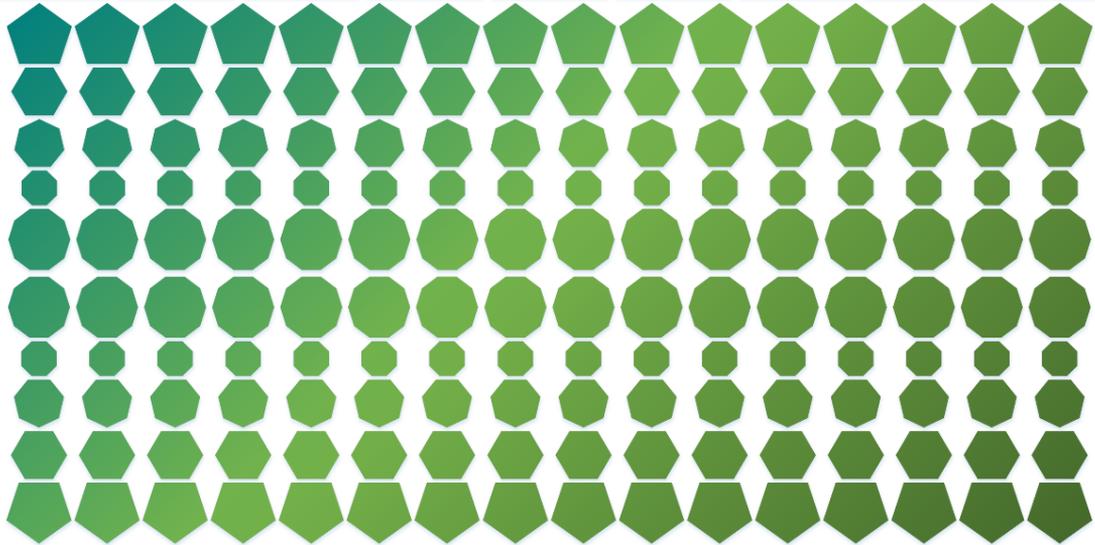
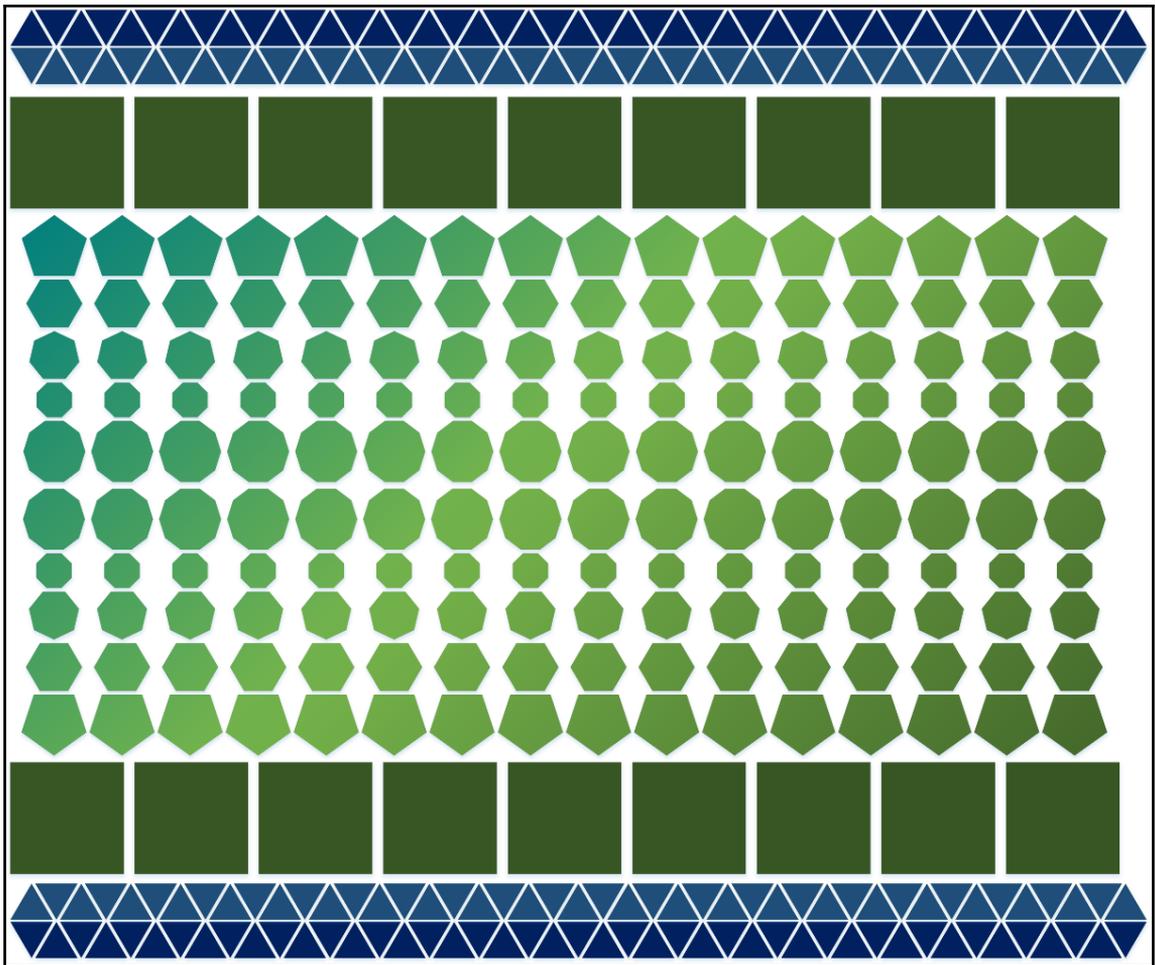
```
1 float calculateRectangleArea(float width, float height) {
2     float area = width * height;
3     System.out.printf("Width: %.2f\n", width);
4     System.out.printf("Height: %.2f\n", height);
5     System.out.printf("Area: %.2f\n", area);
6     return area;
7 }
8
```

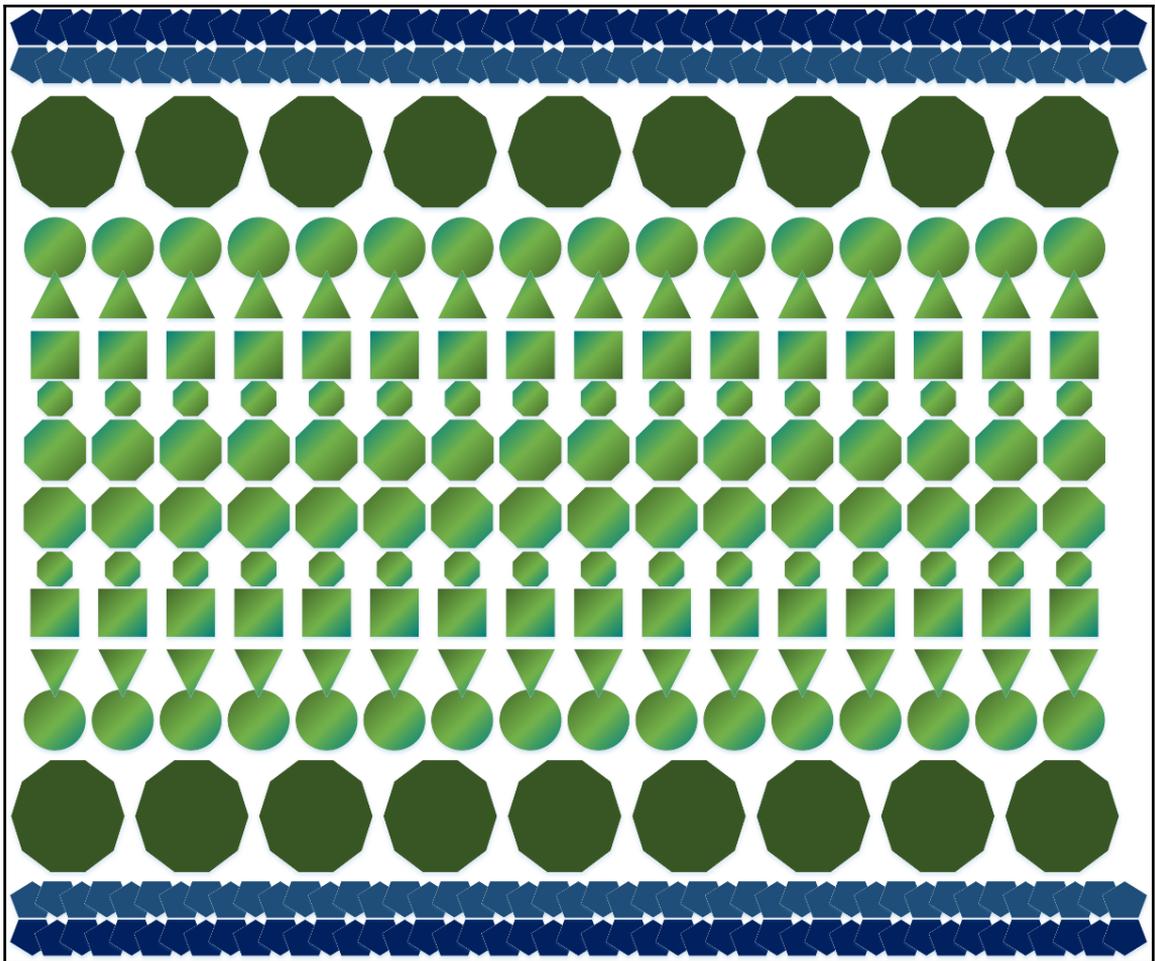
The IDE interface includes a status bar at the bottom with "Line 1, Column 1", "Spaces: 4", and "Java".

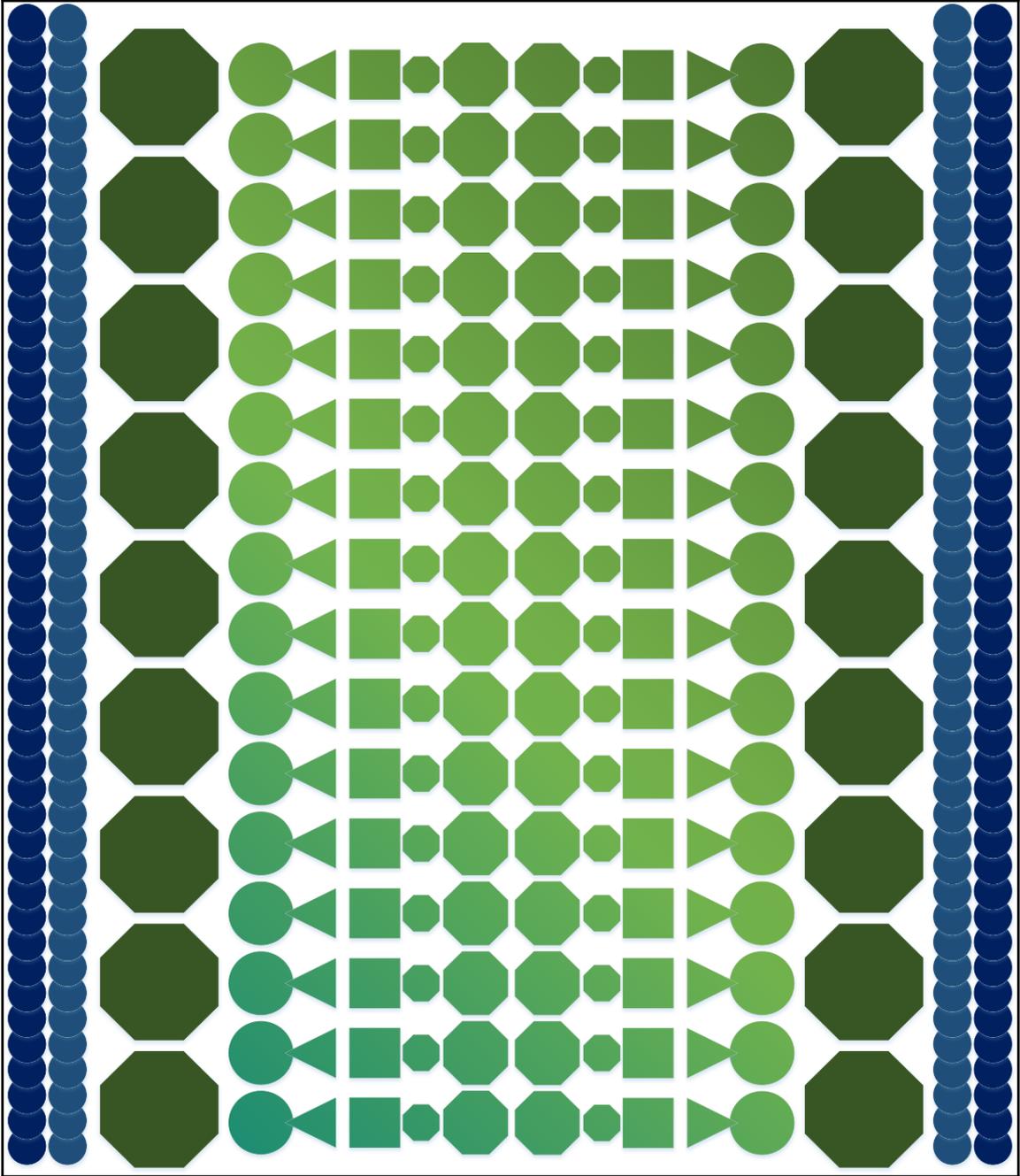
## Chapter 2: Real-World Objects to UML Diagrams and Java 9 via JShell



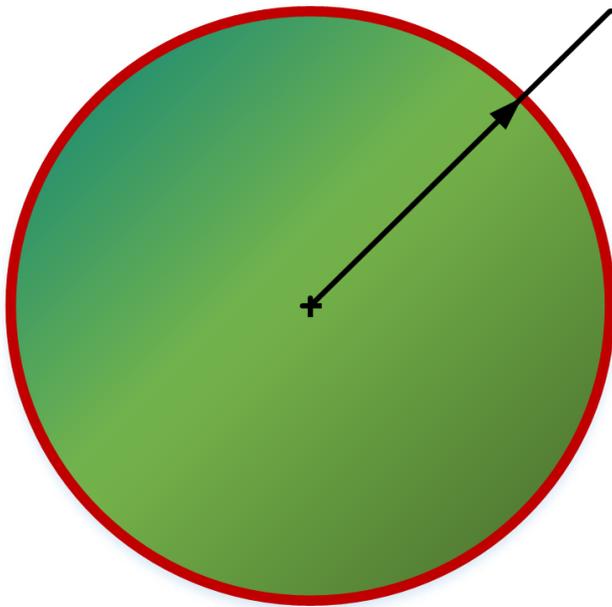








# Circle

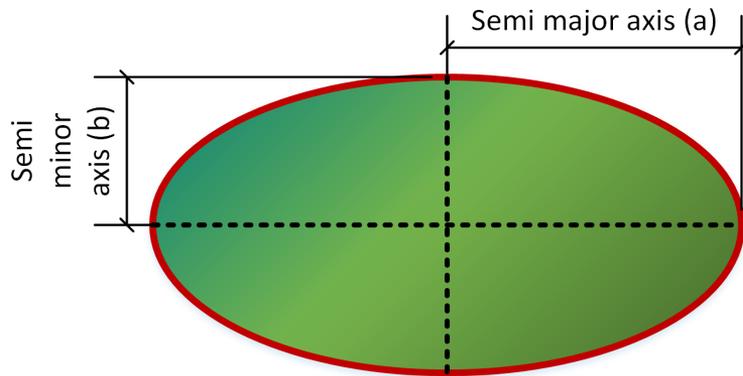


Radius (r)

$$\text{area} = \pi r^2$$

$$\text{perimeter} = \pi 2r$$

## Ellipse

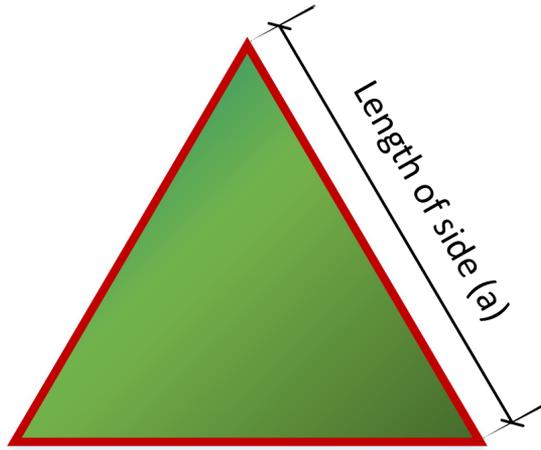


area =  $\pi a b$

perimeter (not so  
accurate

approximation)  $\approx 2\pi \sqrt{\frac{a^2 + b^2}{2}}$

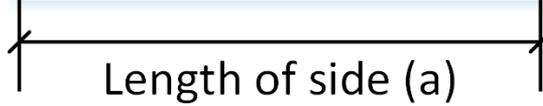
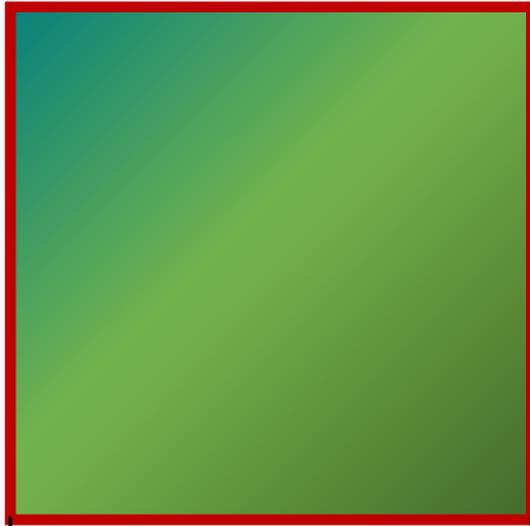
## Equilateral triangle



$$\text{area} = \frac{1\sqrt{3}}{4} a^2$$

$$\text{perimeter} = 3a$$

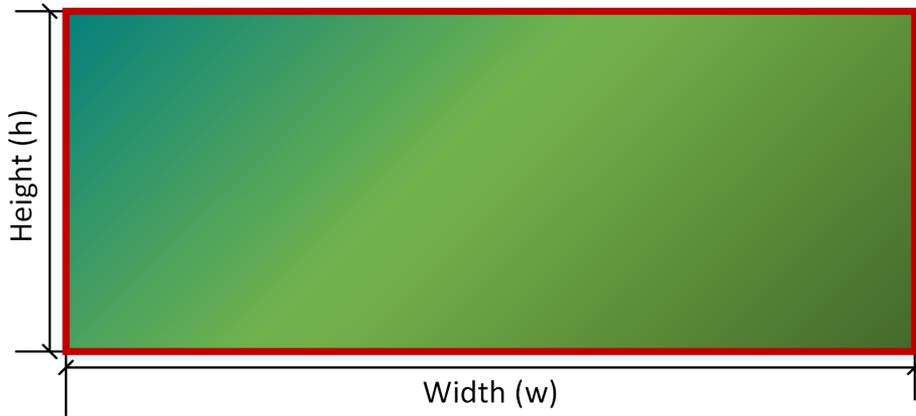
# Square



$$\text{area} = a^2$$

$$\text{perimeter} = 4a$$

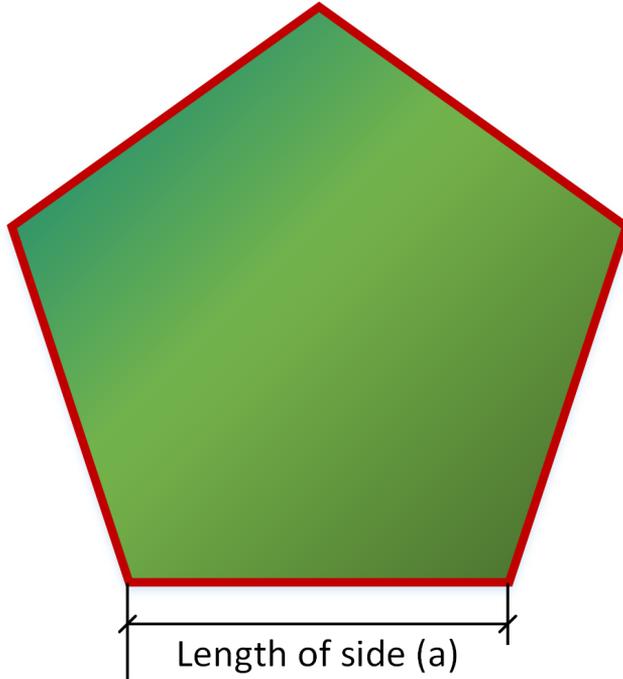
## Rectangle



$$\text{area} = wh$$

$$\text{perimeter} = 2(w+h)$$

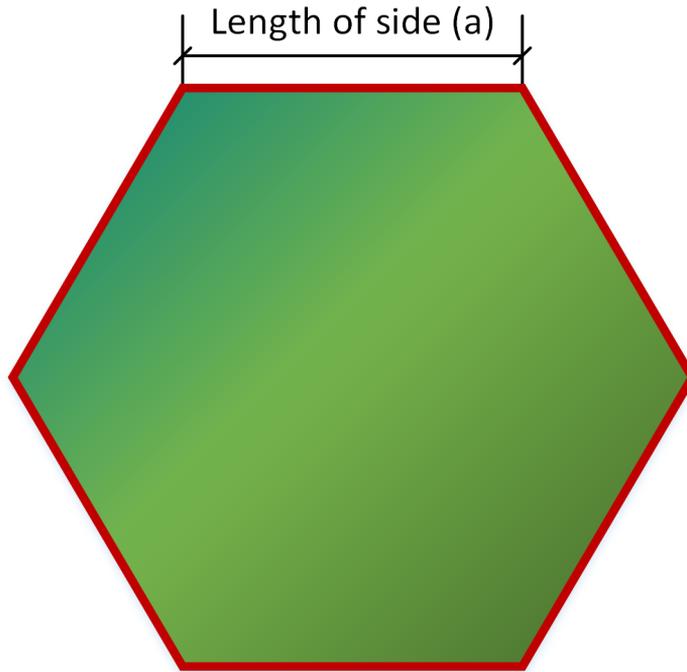
## Regular pentagon



$$\text{area} = \frac{1}{4} \sqrt{5 (5 + 2\sqrt{5})} a^2$$

$$\text{perimeter} = 5a$$

## Regular hexagon

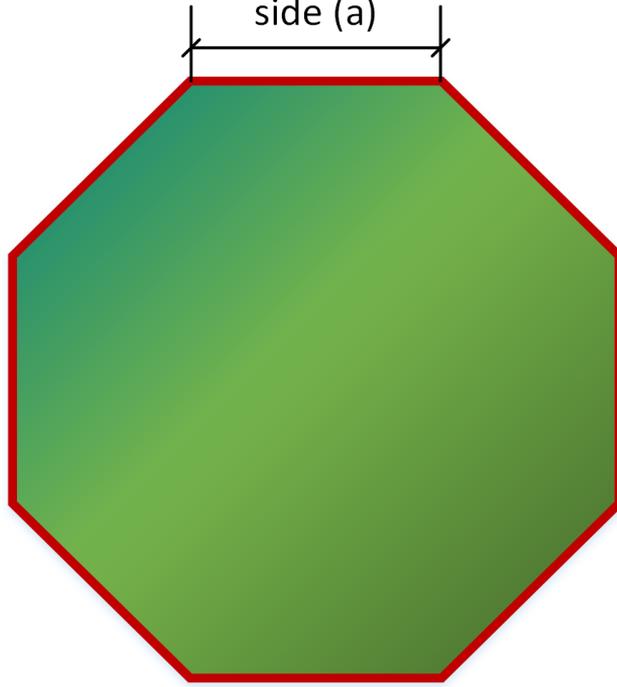


$$\text{area} = \frac{3\sqrt{3}}{2} a^2$$

$$\text{perimeter} = 6a$$

## Regular octagon

Length of  
side (a)

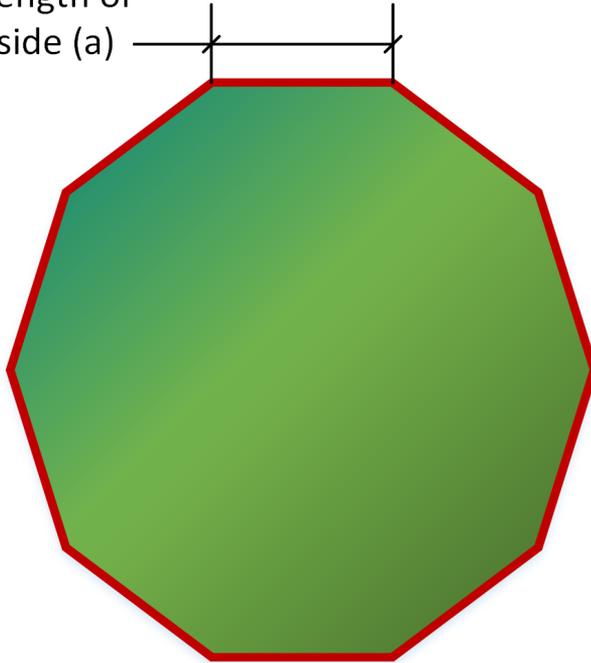


$$\text{area} = 2 (1 + \sqrt{2}) a^2$$

$$\text{perimeter} = 8a$$

# Regular decagon

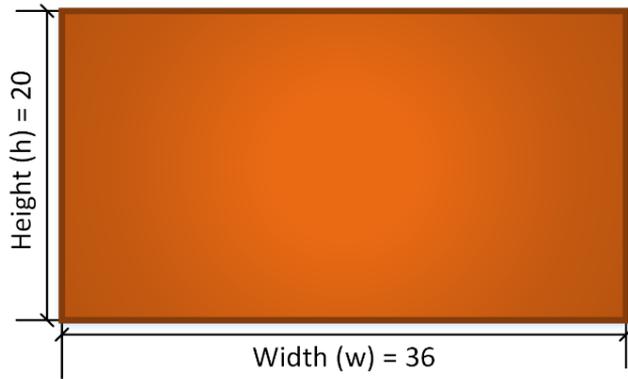
Length of  
side (a)



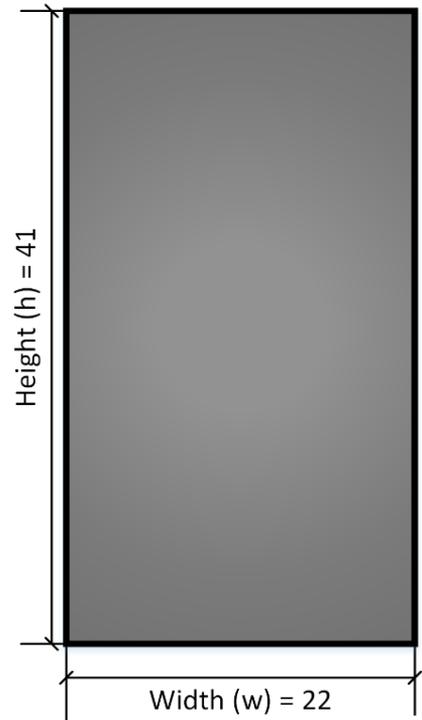
$$\text{area} = \frac{5}{2} a^2 \sqrt{5 + 2\sqrt{5}}$$

$$\text{perimeter} = 10a$$

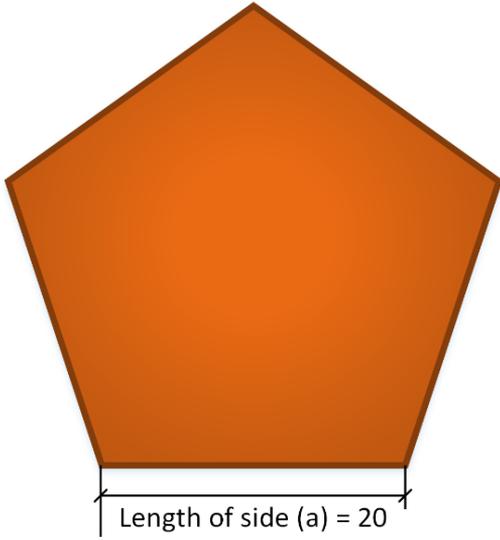
rectangle1



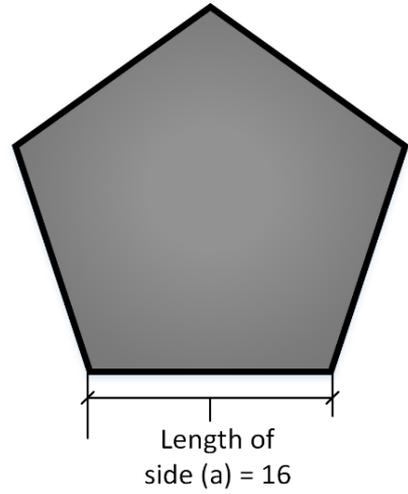
rectangle2



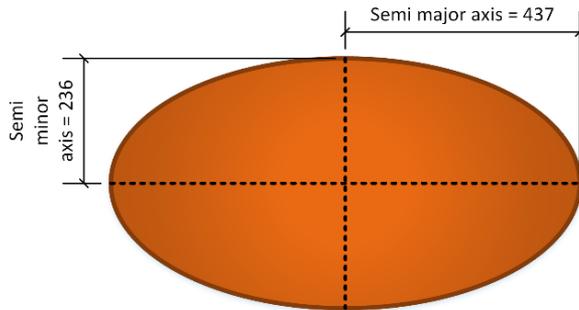
pentagon1



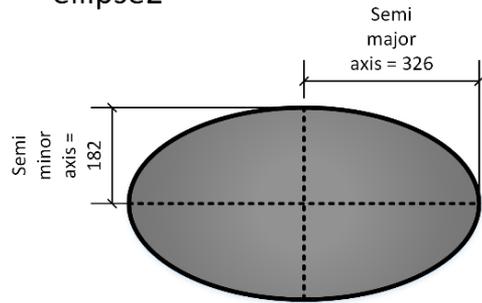
pentagon2



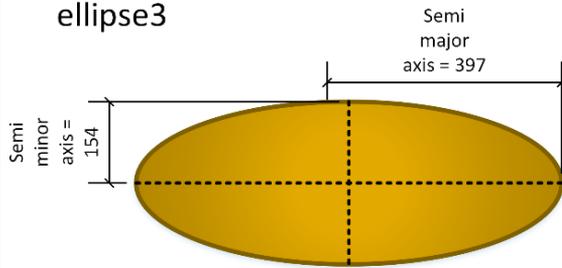
ellipse1



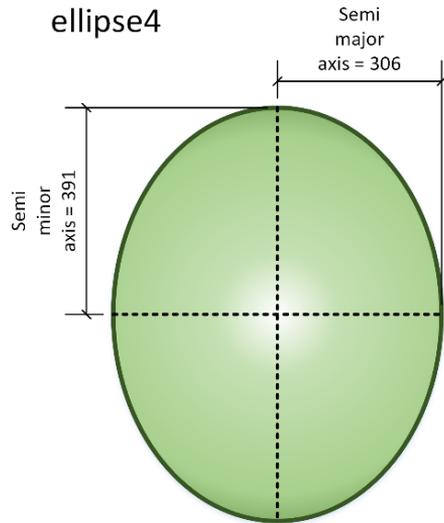
ellipse2



ellipse3



ellipse4



Circle

+radius

Ellipse

+semiMinorAxis  
+semiMajorAxis

EquilateralTriangle

+lengthOfSide

Square

+lengthOfSide

Rectangle

+width  
+height

RegularPentagon

+lengthOfSide

RegularHexagon

+lengthOfSide

RegularOctagon

+lengthOfSide

RegularDecagon

+lengthOfSide

### Circle

+ radius  
-----  
+ calculateArea()  
+ calculatePerimeter()

### Ellipse

+ semiMinorAxis  
+ semiMajorAxis  
-----  
+ calculateArea()  
+ calculatePerimeter()

### EquilateralTriangle

+ lengthOfSide  
-----  
+ calculateArea()  
+ calculatePerimeter()

### Square

+ lengthOfSide  
-----  
+ calculateArea()  
+ calculatePerimeter()

### Rectangle

+ width  
+ height  
-----  
+ calculateArea()  
+ calculatePerimeter()

### RegularPentagon

+ lengthOfSide  
-----  
+ calculateArea()  
+ calculatePerimeter()

### RegularHexagon

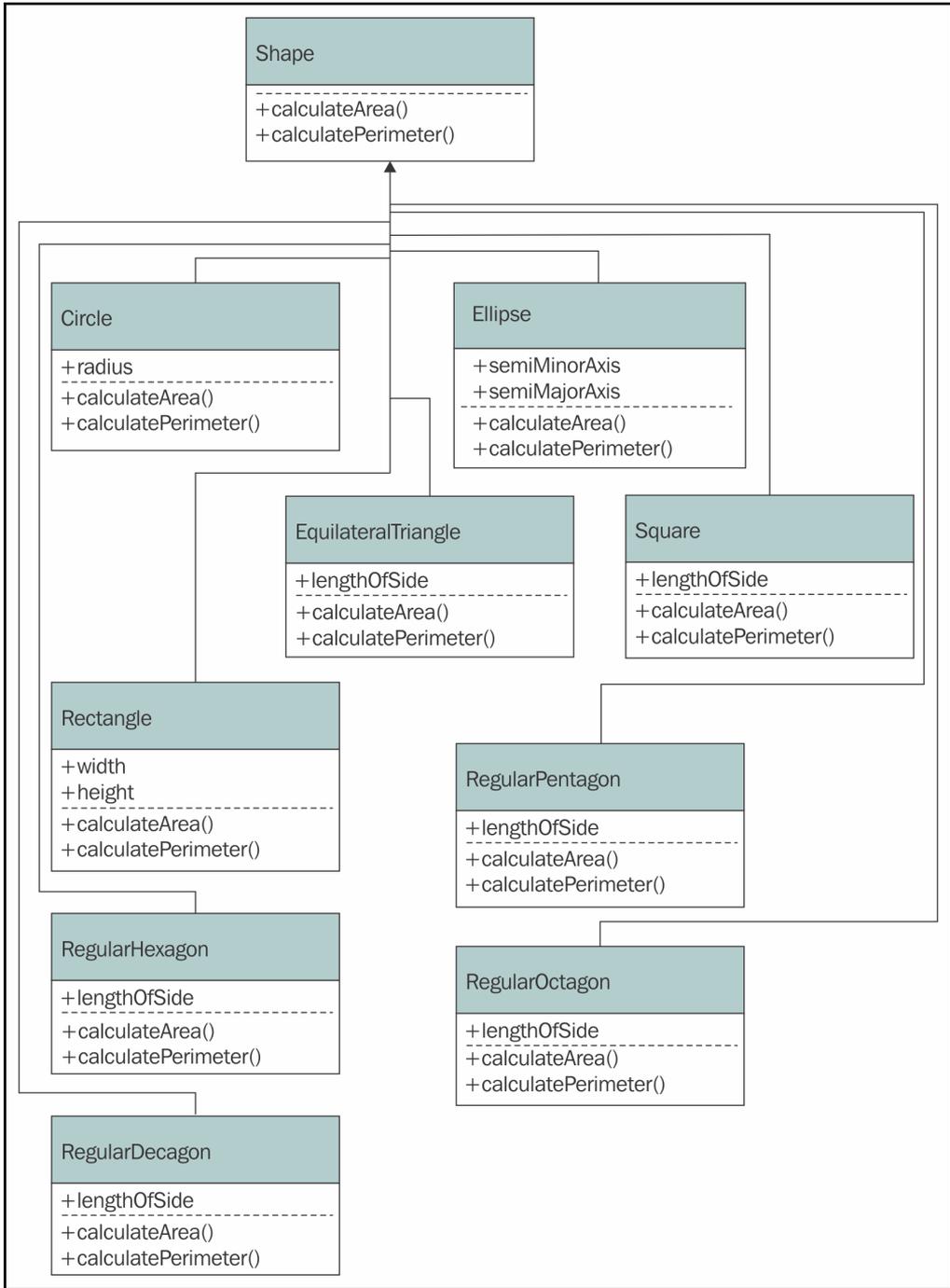
+ lengthOfSide  
-----  
+ calculateArea()  
+ calculatePerimeter()

### RegularOctagon

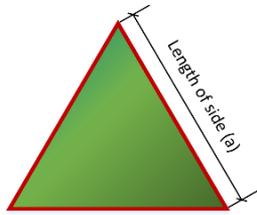
+ lengthOfSide  
-----  
+ calculateArea()  
+ calculatePerimeter()

### RegularDecagon

+ lengthOfSide  
-----  
+ calculateArea()  
+ calculatePerimeter()

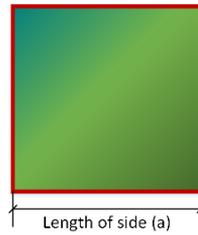


### Equilateral triangle



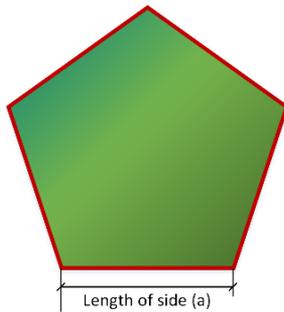
Number of sides (n) = 3

### Square



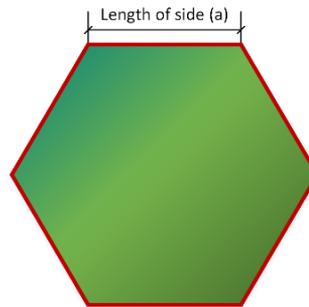
Number of sides (n) = 4

### Regular pentagon



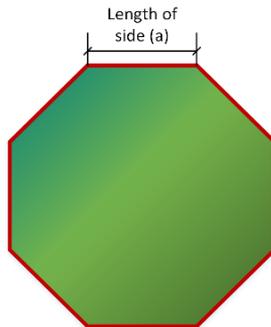
Number of sides (n) = 5

### Regular hexagon



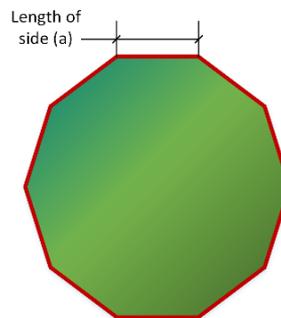
Number of sides (n) = 6

### Regular octagon



Number of sides (n) = 8

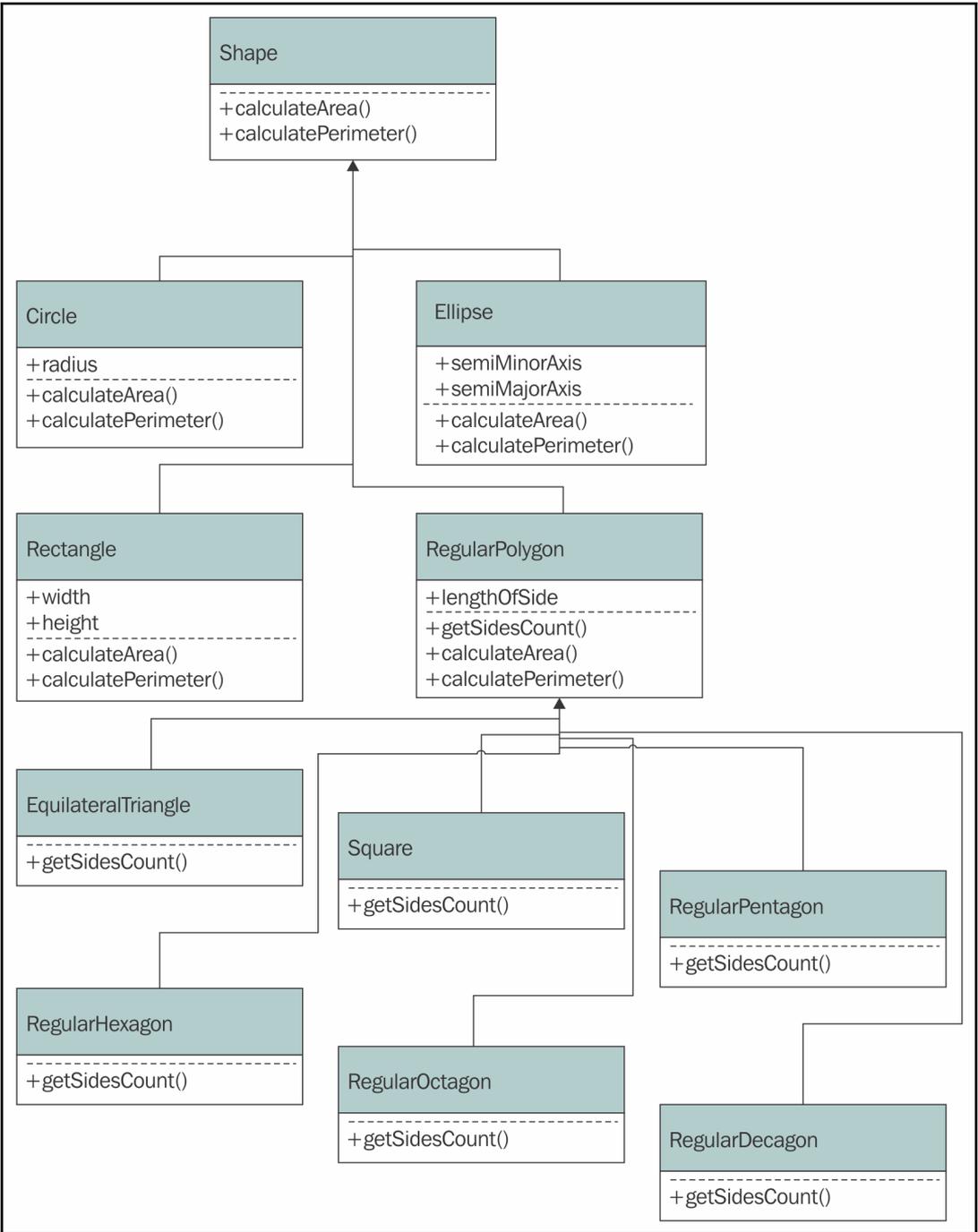
### Regular decagon

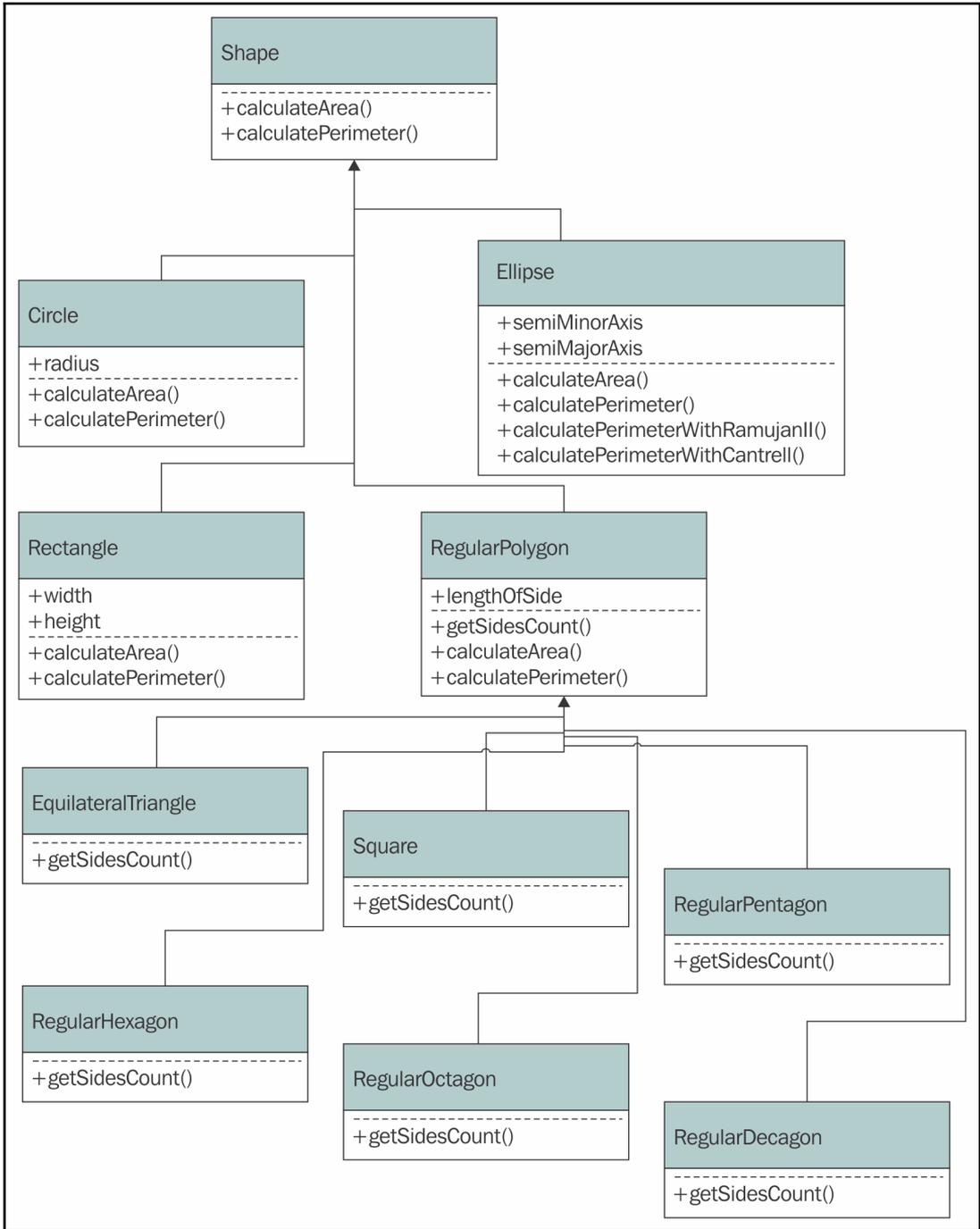


Number of sides (n) = 10

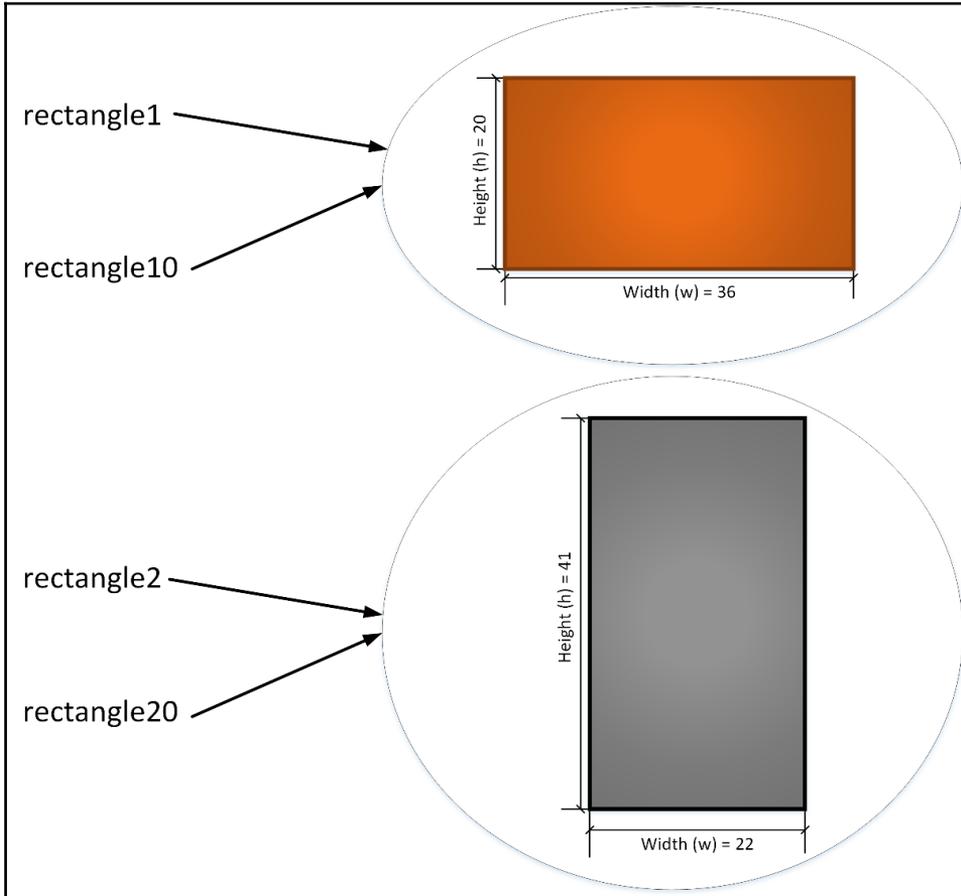
$$\text{area} = \frac{1}{4} na^2 \cot\left(\frac{\pi}{n}\right)$$

$$\text{perimeter} = na$$



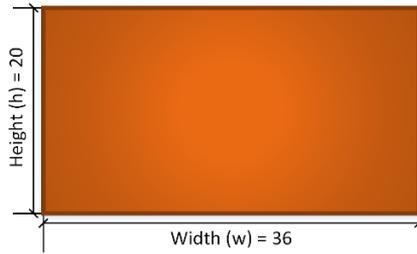


# Chapter 3: Classes and Instances



rectangle1 

rectangle10



rectangle2 

rectangle20

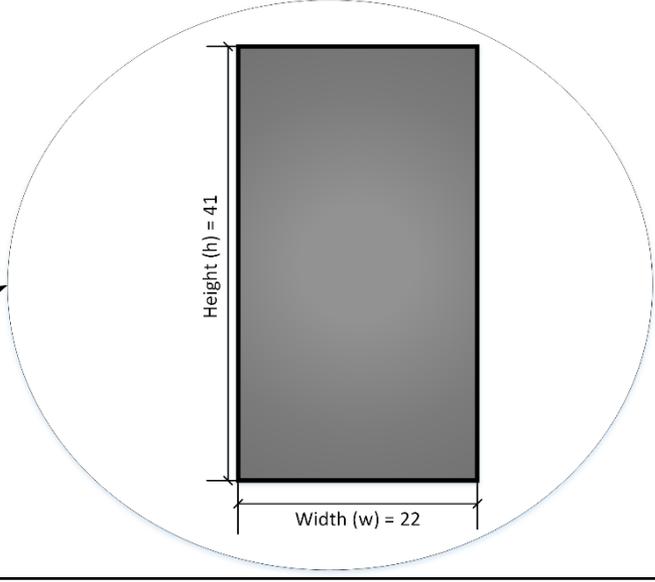
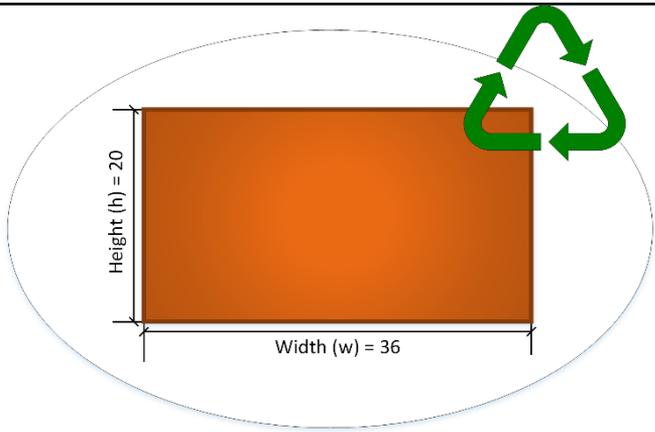


rectangle1 

rectangle10 

rectangle2 

rectangle20 

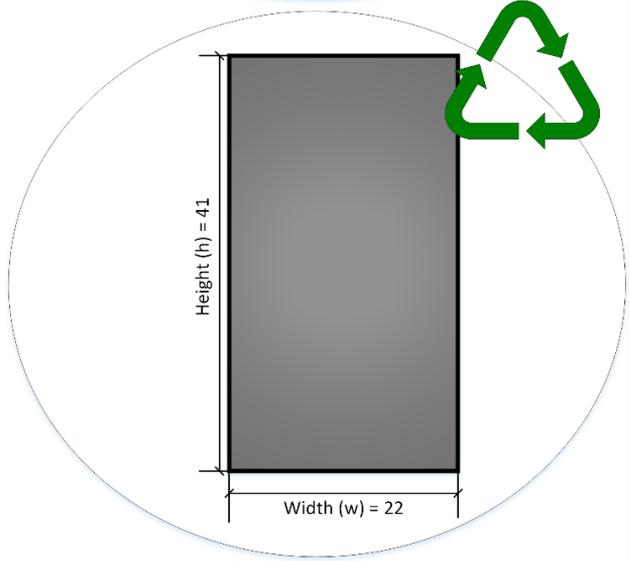
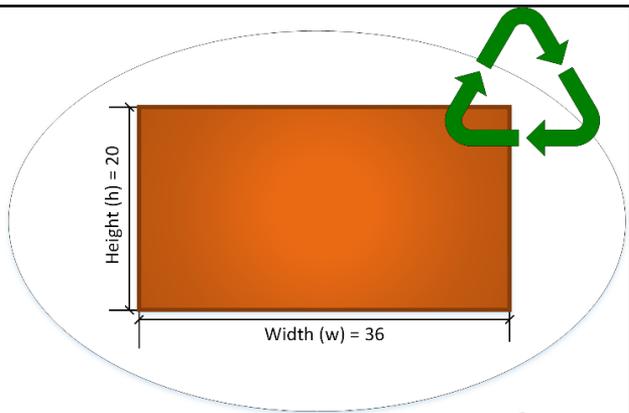


rectangle1 

rectangle10 

rectangle2 

rectangle20 



```
jshell> class Rectangle {
...>     double width;
...>     double height;
...>
...>     Rectangle(double width, double height) {
...>         System.out.printf("Initializing a new Rectangle instance\n");
...>         System.out.printf("Width: %.2f, Height: %.2f\n", width, height);
...>         this.width = width;
...>         this.height = height;
...>     }
...> }
| created class Rectangle

jshell>

jshell> Rectangle rectangle1 = new Rectangle(31.0, 21.0);
Initializing a new Rectangle instance
Width: 31.00, Height: 21.00
rectangle1 ==> Rectangle@551aa95a

jshell> Rectangle rectangle2 = new Rectangle(182.0, 32.0);
Initializing a new Rectangle instance
Width: 182.00, Height: 32.00
rectangle2 ==> Rectangle@1dfe2924

jshell> Rectangle rectangle3 = new Rectangle(203.0, 23.0);
Initializing a new Rectangle instance
Width: 203.00, Height: 23.00
rectangle3 ==> Rectangle@6e6c3152

jshell> Rectangle rectangle4 = new Rectangle(404.0, 14.0);
Initializing a new Rectangle instance
Width: 404.00, Height: 14.00
rectangle4 ==> Rectangle@3cef309d

jshell>
```

```
jshell>  
  
jshell> rectangle1.width  
$6 ==> 31.0  
  
jshell> rectangle1.height  
$7 ==> 21.0  
  
jshell> rectangle2.width  
$8 ==> 182.0  
  
jshell> rectangle2.height  
$9 ==> 32.0  
  
jshell> rectangle3.width  
$10 ==> 203.0  
  
jshell> rectangle3.height  
$11 ==> 23.0  
  
jshell> rectangle4.width  
$12 ==> 404.0  
  
jshell> rectangle4.height  
$13 ==> 14.0  
  
jshell>
```

```
jshell> Rectangle rectangleError = new Rectangle();
| Error:
| constructor Rectangle in class Rectangle cannot be applied to given types;
|   required: double,double
|   found: no arguments
|   reason: actual and formal argument lists differ in length
| Rectangle rectangleError = new Rectangle();
|                               ^-----^
|
jshell>
```

```
jshell> Rectangle rectangleToCollect1 = new Rectangle(51, 121);
| Initializing a new Rectangle instance
| Width: 51.00, Height: 121.00
| rectangleToCollect1 ==> Rectangle@551aa95a
|
jshell> Rectangle rectangleToCollect2 = new Rectangle(72, 282);
| Initializing a new Rectangle instance
| Width: 72.00, Height: 282.00
| rectangleToCollect2 ==> Rectangle@1dfe2924
|
jshell> rectangleToCollect1 = null;
| rectangleToCollect1 ==> null
|
jshell> rectangleToCollect2 = null;
| rectangleToCollect2 ==> null
|
jshell>
```

```
jshell> rectangleToCollect1 = null;  
rectangleToCollect1 ==> null  
  
jshell> rectangleToCollect2 = null;  
rectangleToCollect2 ==> null  
  
jshell> System.gc();  
Finalizing Rectangle  
  
jshell> Width: 72.00, Height: 282.00  
Finalizing Rectangle  
Width: 51.00, Height: 121.00  
  
jshell>
```

```
jshell> Rectangle rectangle5 = new Rectangle(50, 550);
Initializing a new Rectangle instance
Width: 50.00, Height: 550.00
rectangle5 ==> Rectangle@34c4973

jshell> Rectangle referenceToRectangle5 = rectangle5;
referenceToRectangle5 ==> Rectangle@34c4973

jshell> rectangle5 = null;
rectangle5 ==> null

jshell> System.gc();

jshell>
```

```
jshell> System.gc();

jshell> referenceToRectangle5 = null;
referenceToRectangle5 ==> null

jshell> System.gc();
Finalizing Rectangle
Width: 50.00, Height: 550.00
jshell>
```

```
jshell> double getGeneratedRectangleHeight() {
...>     final Rectangle rectangle = new Rectangle(37, 87);
...>     return rectangle.height;
...> }
| created method getGeneratedRectangleHeight()

jshell>

jshell> System.out.printf("Height: %.2f\n", getGeneratedRectangleHeight());
Initializing a new Rectangle instance
Width: 37.00, Height: 87.00
Height: 87.00
$30 ==> java.io.PrintStream@335eadca
| created scratch variable $30 : PrintStream

jshell> System.gc();
Finalizing Rectangle
Width: 37.00
jshell> , Height:
87.00
jshell>
```

## Chapter 4: Encapsulation of Data

```
jshell> class VirtualCreature {
...>     String name;
...>     int birthYear;
...>
...>     VirtualCreature(String name, int birthYear) {
...>         this.name = name;
...>         this.birthYear = birthYear;
...>     }
...> }
| created class VirtualCreature

jshell> VirtualCreature beedrill = new VirtualCreature("Beedrill", 2014);
beedrill ==> VirtualCreature@4b9e13df
| created variable beedrill : VirtualCreature

jshell> System.out.printf("%s\n", beedrill.name);
Beedrill
$3 ==> java.io.PrintStream@d8355a8
| created scratch variable $3 : PrintStream

jshell> System.out.printf("%d\n", beedrill.birthYear);
2014
$4 ==> java.io.PrintStream@d8355a8
| created scratch variable $4 : PrintStream

jshell> VirtualCreature krabby = new VirtualCreature("Krabby", 2012);
krabby ==> VirtualCreature@4501b7af
| created variable krabby : VirtualCreature

jshell> System.out.printf("%s\n", krabby.name);
Krabby
$6 ==> java.io.PrintStream@d8355a8
| created scratch variable $6 : PrintStream

jshell> System.out.printf("%d\n", krabby.birthYear);
2012
$7 ==> java.io.PrintStream@d8355a8
| created scratch variable $7 : PrintStream
```

```
jshell> class VirtualCreature {
...>     final String name;
...>     final int birthYear;
...>
...>     VirtualCreature(String name, int birthYear) {
...>         this.name = name;
...>         this.birthYear = birthYear;
...>     }
...> }
| created class VirtualCreature

jshell> VirtualCreature squirtle = new VirtualCreature("Squirtle", 2014);
squirtle ==> VirtualCreature@4b9e13df
| created variable squirtle : VirtualCreature

jshell> System.out.printf("%s\n", squirtle.name);
Squirtle
$3 ==> java.io.PrintStream@d8355a8
| created scratch variable $3 : PrintStream

jshell> System.out.printf("%d\n", squirtle.birthYear);
2014
$4 ==> java.io.PrintStream@d8355a8
| created scratch variable $4 : PrintStream

jshell> squirtle.name = "Tentacrue1";
| Error:
| cannot assign a value to final variable name
| squirtle.name = "Tentacrue1";
| ^-----^

jshell> squirtle.birthYear = 2017;
| Error:
| cannot assign a value to final variable birthYear
| squirtle.birthYear = 2017;
| ^-----^
```

```
jshell> import java.time.Year;

jshell>

jshell> class VirtualCreature {
...>     final String name;
...>     final int birthYear;
...>
...>     VirtualCreature(String name, int birthYear) {
...>         this.name = name;
...>         this.birthYear = birthYear;
...>     }
...>
...>     int getAge() {
...>         return Year.now().getValue() - birthYear;
...>     }
...> }
| created class VirtualCreature

jshell>

jshell> VirtualCreature arbok = new VirtualCreature("Arbok", 2008);
arbok ==> VirtualCreature@4b9e13df
| created variable arbok : VirtualCreature

jshell> System.out.printf("%d\n", arbok.getAge());
9
$4 ==> java.io.PrintStream@4e1d422d
| created scratch variable $4 : PrintStream

jshell> VirtualCreature pidgey = new VirtualCreature("Pidgey", 2015);
pidgey ==> VirtualCreature@52a86356
| created variable pidgey : VirtualCreature

jshell> System.out.printf("%d\n", pidgey.getAge());
2
$6 ==> java.io.PrintStream@4e1d422d
| created scratch variable $6 : PrintStream
```

```
...>         birthYear = Year.now().getValue() - age;
...>     }
...> }
| created class VirtualCreature

jshell> VirtualCreature venusaur = new VirtualCreature("Venusaur", 2000);
venusaur ==> VirtualCreature@4b9e13df

jshell> System.out.printf("%d\n", venusaur.getAge());
17
$4 ==> java.io.PrintStream@4e1d422d

jshell> VirtualCreature caterpie = new VirtualCreature("Caterpie", 2012);
caterpie ==> VirtualCreature@52a86356

jshell> System.out.printf("%d\n", caterpie.getAge());
5
$6 ==> java.io.PrintStream@4e1d422d

jshell> venusaur.setAge(2);

jshell> System.out.printf("%d\n", venusaur.getAge());
2
$8 ==> java.io.PrintStream@4e1d422d

jshell> System.out.printf("%d\n", venusaur.birthYear);
2015
$9 ==> java.io.PrintStream@4e1d422d

jshell> venusaur.setAge(14);

jshell> System.out.printf("%d\n", caterpie.getAge());
5
$11 ==> java.io.PrintStream@4e1d422d

jshell> System.out.printf("%d\n", caterpie.birthYear);
2012
$12 ==> java.io.PrintStream@4e1d422d
```

```
jshell> VirtualCreature persian = new VirtualCreature("Persian", 2005);
persian ==> VirtualCreature@4b9e13df
| created variable persian : VirtualCreature

jshell> System.out.printf("%d\n", persian.getAge());
12
$4 ==> java.io.PrintStream@4e1d422d
| created scratch variable $4 : PrintStream

jshell> VirtualCreature arcanine = new VirtualCreature("Arcanine", 2012);
arcanine ==> VirtualCreature@52a86356
| created variable arcanine : VirtualCreature

jshell> System.out.printf("%d\n", arcanine.getAge());
5
$6 ==> java.io.PrintStream@4e1d422d
| created scratch variable $6 : PrintStream

jshell> persian.setAge(7);

jshell> System.out.printf("%d\n", persian.getAge());
7
$8 ==> java.io.PrintStream@4e1d422d
| created scratch variable $8 : PrintStream

jshell> System.out.printf("%d\n", persian.birthYear);
2010
$9 ==> java.io.PrintStream@4e1d422d
| created scratch variable $9 : PrintStream

jshell> arcanine.setAge(9);

jshell> System.out.printf("%d\n", arcanine.getAge());
9
$11 ==> java.io.PrintStream@4e1d422d
| created scratch variable $11 : PrintStream

jshell> System.out.printf("%d\n", arcanine.birthYear);
2008
```

```
jshell> VirtualCreature glaceon = new VirtualCreature("Glaceon", 2009, "Baseball cap");
glaceon ==> VirtualCreature@4b9e13df
| created variable glaceon : VirtualCreature

jshell> System.out.printf(glaceon.getHat());
BASEBALL CAP$4 ==> java.io.PrintStream@d8355a8
| created scratch variable $4 : PrintStream

jshell> glaceon.setHat("Hard hat")

jshell> System.out.printf(glaceon.getHat());
HARD HAT$6 ==> java.io.PrintStream@d8355a8
| created scratch variable $6 : PrintStream

jshell> VirtualCreature gliscor = new VirtualCreature("Gliscor", 2015, "Cowboy hat");
gliscor ==> VirtualCreature@523884b2
| created variable gliscor : VirtualCreature

jshell> System.out.printf(gliscor.getHat());
COWBOY HAT$8 ==> java.io.PrintStream@d8355a8
| created scratch variable $8 : PrintStream

jshell> gliscor.setHat("Panama hat")

jshell> System.out.printf(gliscor.getHat());
PANAMA HAT$10 ==> java.io.PrintStream@d8355a8
| created scratch variable $10 : PrintStream
```

```
jshell> System.out.printf(gliscor.hat);
| Error:
| hat has private access in VirtualCreature
| System.out.printf(gliscor.hat);
|                          ^-----^

jshell> System.out.printf("%d", glaceon.getCurrentYear());
| Error:
| getCurrentYear() has private access in VirtualCreature
| System.out.printf("%d", glaceon.getCurrentYear());
|                          ^-----^

jshell>
```

```
jshell> VirtualCreature lairon =
...>     new VirtualCreature("Lairon", 2014, "Sombrero", 150);
lairon ==> VirtualCreature@4b9e13df
| created variable lairon : VirtualCreature

jshell> System.out.printf("%d", lairon.getVisibilityLevel());
100$4 ==> java.io.PrintStream@d8355a8
| created scratch variable $4 : PrintStream

jshell> lairon.setVisibilityLevel(-6);

jshell> System.out.printf("%d", lairon.getVisibilityLevel());
0$6 ==> java.io.PrintStream@d8355a8
| created scratch variable $6 : PrintStream

jshell> lairon.setVisibilityLevel(320);

jshell> System.out.printf("%d", lairon.getVisibilityLevel());
100$8 ==> java.io.PrintStream@d8355a8
| created scratch variable $8 : PrintStream

jshell> lairon.setVisibilityLevel(25);

jshell> System.out.printf("%d", lairon.getVisibilityLevel());
25$10 ==> java.io.PrintStream@d8355a8
| created scratch variable $10 : PrintStream

jshell>
```

```
jshell> System.out.printf("%d\n", VirtualCreature.SPECIAL_ATTACK_POWER);
35
$3 ==> java.io.PrintStream@14acaea5
| created scratch variable $3 : PrintStream

jshell> System.out.printf("%d\n", VirtualCreature.SPECIAL_DEFENSE_POWER);
95
$4 ==> java.io.PrintStream@14acaea5
| created scratch variable $4 : PrintStream

jshell> VirtualCreature golbat =
...> new VirtualCreature("Golbat", 2015, "Baseball cap", 75);
golbat ==> VirtualCreature@59fa1d9b
| created variable golbat : VirtualCreature

jshell> System.out.printf("%d\n", golbat.GROWTH_RATE);
10
$6 ==> java.io.PrintStream@14acaea5
| created scratch variable $6 : PrintStream

jshell>
```

```
jshell> System.out.printf("%d\n", VirtualCreature.getSpecialAttackPower());
35
$3 ==> java.io.PrintStream@14acaea5
| created scratch variable $3 : PrintStream

jshell> System.out.printf("%d\n", VirtualCreature.getSpecialDefensePower());
95
$4 ==> java.io.PrintStream@14acaea5
| created scratch variable $4 : PrintStream

jshell> VirtualCreature vulpix =
...> new VirtualCreature("Vulpix", 2012, "Fedora", 35);
vulpix ==> VirtualCreature@59fa1d9b
| created variable vulpix : VirtualCreature

jshell> System.out.printf("%d", vulpix.getGrowthRate())
10$6 ==> java.io.PrintStream@14acaea5
| created scratch variable $6 : PrintStream
```

## Chapter 5: Mutable and Immutable Classes

```
jshell> Vector3d vector1 = new Vector3d(10.0, 20.0, 30.0);
vector1 ==> (x: 10.00, y: 20.00, z: 30.00)
| created variable vector1 : Vector3d

jshell> Vector3d vector2 = new Vector3d(1.0, 2.0, 3.0);
vector2 ==> (x: 1.00, y: 2.00, z: 3.00)
| created variable vector2 : Vector3d

jshell> System.out.println(vector1);
(x: 10.00, y: 20.00, z: 30.00)

jshell> System.out.println(vector2);
(x: 1.00, y: 2.00, z: 3.00)

jshell> vector1.add(vector2);

jshell> System.out.println(vector1);
(x: 11.00, y: 22.00, z: 33.00)

jshell>
```

```
jshell> Vector3d vector3 = new Vector3d();
vector3 ==> (x: 0.00, y: 0.00, z: 0.00)
| created variable vector3 : Vector3d

jshell> Vector3d vector4 = new Vector3d(5.0);
vector4 ==> (x: 5.00, y: 5.00, z: 5.00)
| created variable vector4 : Vector3d

jshell> Vector3d vector5 = new Vector3d(-15.5, -11.1, -8.8);
vector5 ==> (x: -15.50, y: -11.10, z: -8.80)
| created variable vector5 : Vector3d

jshell> System.out.println(vector3);
(x: 0.00, y: 0.00, z: 0.00)

jshell> System.out.println(vector4);
(x: 5.00, y: 5.00, z: 5.00)

jshell> System.out.println(vector5);
(x: -15.50, y: -11.10, z: -8.80)
```

```
jshell> vector4.negate();

jshell> System.out.println(vector4);
(x: -5.00, y: -5.00, z: -5.00)

jshell> vector3.add(vector4);

jshell> System.out.println(vector3);
(x: -5.00, y: -5.00, z: -5.00)

jshell> vector4.absolute();

jshell> System.out.println(vector4);
(x: 5.00, y: 5.00, z: 5.00)

jshell> vector5.sub(vector4);

jshell> System.out.println(vector5);
(x: -20.50, y: -16.10, z: -13.80)

jshell>
```

```
jshell> ImmutableVector3d vector10 =  
  ...>      new ImmutableVector3d(100.0, 200.0, 300.0);  
vector10 ==> (x: 100.00, y: 200.00, z: 300.00)  
| created variable vector10 : ImmutableVector3d  
  
jshell> ImmutableVector3d vector20 =  
  ...>      new ImmutableVector3d(11.0, 12.0, 13.0);  
vector20 ==> (x: 11.00, y: 12.00, z: 13.00)  
| created variable vector20 : ImmutableVector3d  
  
jshell> System.out.println(vector10);  
(x: 100.00, y: 200.00, z: 300.00)  
  
jshell> System.out.println(vector20);  
(x: 11.00, y: 12.00, z: 13.00)  
  
jshell> ImmutableVector3d vector30 = vector10.add(vector20);  
vector30 ==> (x: 111.00, y: 212.00, z: 313.00)  
| created variable vector30 : ImmutableVector3d  
  
jshell> System.out.println(vector30);  
(x: 111.00, y: 212.00, z: 313.00)  
  
jshell>
```

```
jshell> ImmutableVector3d vector40 = new ImmutableVector3d();
vector40 ==> (x: 0.00, y: 0.00, z: 0.00)
| created variable vector40 : ImmutableVector3d

jshell> ImmutableVector3d vector50 = new ImmutableVector3d(-5.0);
vector50 ==> (x: -5.00, y: -5.00, z: -5.00)
| created variable vector50 : ImmutableVector3d

jshell> ImmutableVector3d vector60 = new ImmutableVector3d(8.0, 9.0, 10.0);
vector60 ==> (x: 8.00, y: 9.00, z: 10.00)
| created variable vector60 : ImmutableVector3d

jshell> System.out.println(vector40);
(x: 0.00, y: 0.00, z: 0.00)

jshell> System.out.println(vector50);
(x: -5.00, y: -5.00, z: -5.00)

jshell> System.out.println(vector60);
(x: 8.00, y: 9.00, z: 10.00)

jshell>
```

```
jshell> ImmutableVector3d vector70 = vector50.negate();
vector70 ==> (x: 5.00, y: 5.00, z: 5.00)
| created variable vector70 : ImmutableVector3d

jshell> System.out.println(vector70);
(x: 5.00, y: 5.00, z: 5.00)

jshell> ImmutableVector3d vector80 = vector40.add(vector70);
vector80 ==> (x: 5.00, y: 5.00, z: 5.00)
| created variable vector80 : ImmutableVector3d

jshell> System.out.println(vector80);
(x: 5.00, y: 5.00, z: 5.00)

jshell> ImmutableVector3d vector90 = vector70.absolute();
vector90 ==> (x: 5.00, y: 5.00, z: 5.00)
| created variable vector90 : ImmutableVector3d

jshell> System.out.println(vector90);
(x: 5.00, y: 5.00, z: 5.00)

jshell> ImmutableVector3d vector100 = vector60.sub(vector90);
vector100 ==> (x: 3.00, y: 4.00, z: 5.00)
| created variable vector100 : ImmutableVector3d

jshell> System.out.println(vector100);
(x: 3.00, y: 4.00, z: 5.00)

jshell>
```

```
jshell> Vector3d mutableVector3d1 =
...>     new Vector3d(-30.5, -15.5, -12.5);
mutableVector3d1 ==> (x: -30.50, y: -15.50, z: -12.50)
| created variable mutableVector3d1 : Vector3d

jshell> System.out.println(mutableVector3d1);
(x: -30.50, y: -15.50, z: -12.50)

jshell> mutableVector3d1.absolute();

jshell> System.out.println(mutableVector3d1);
(x: 30.50, y: 15.50, z: 12.50)
```

```
jshell> ImmutableVector3d immutableVector3d1 =
...>     new ImmutableVector3d(-30.5, -15.5, -12.5);
immutableVector3d1 ==> (x: -30.50, y: -15.50, z: -12.50)
| created variable immutableVector3d1 : ImmutableVector3d

jshell> System.out.println(immutableVector3d1);
(x: -30.50, y: -15.50, z: -12.50)

jshell> ImmutableVector3d immutableVector3d2 =
...>     immutableVector3d1.absolute();
immutableVector3d2 ==> (x: 30.50, y: 15.50, z: 12.50)
| created variable immutableVector3d2 : ImmutableVector3d

jshell> System.out.println(immutableVector3d2);
(x: 30.50, y: 15.50, z: 12.50)

jshell>
```

```
jshell> String welcomeMessage = "Welcome to Virtual Creatures Land";
welcomeMessage ==> "Welcome to Virtual Creatures Land"
|   created variable welcomeMessage : String

jshell> System.out.println(welcomeMessage);
Welcome to Virtual Creatures Land

jshell> System.out.println(welcomeMessage.toUpperCase());
WELCOME TO VIRTUAL CREATURES LAND

jshell> System.out.println(welcomeMessage.toLowerCase());
welcome to virtual creatures land

jshell> System.out.println(welcomeMessage.replaceAll(" ", "-"));
Welcome-to-Virtual-Creatures-Land

jshell> System.out.println(welcomeMessage);
Welcome to Virtual Creatures Land

jshell>
```

```
jshell> ImmutableVirtualCreature meowth1 =
...>     new ImmutableVirtualCreature(
...>         "Meowth", 2010, "Baseball cap", 35);
meowth1 ==> ImmutableVirtualCreature@4b9e13df
| created variable meowth1 : ImmutableVirtualCreature

jshell> ImmutableVirtualCreature meowth2 =
...>     meowth1.evolveToAge(3);
meowth2 ==> ImmutableVirtualCreature@31dc339b
| created variable meowth2 : ImmutableVirtualCreature

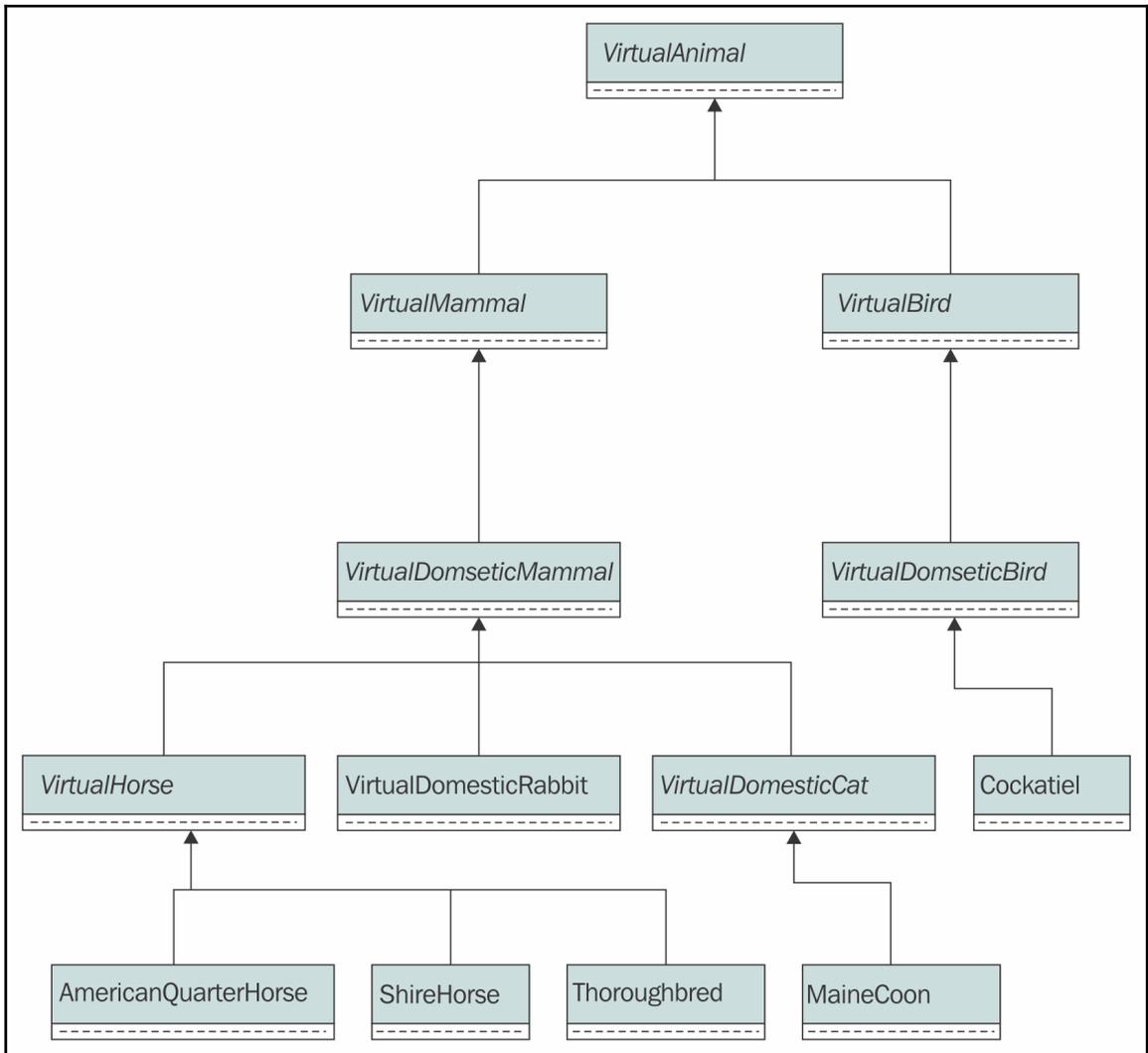
jshell> System.out.printf("%d\n", meowth2.getAge());
3
$5 ==> java.io.PrintStream@52a86356
| created scratch variable $5 : PrintStream

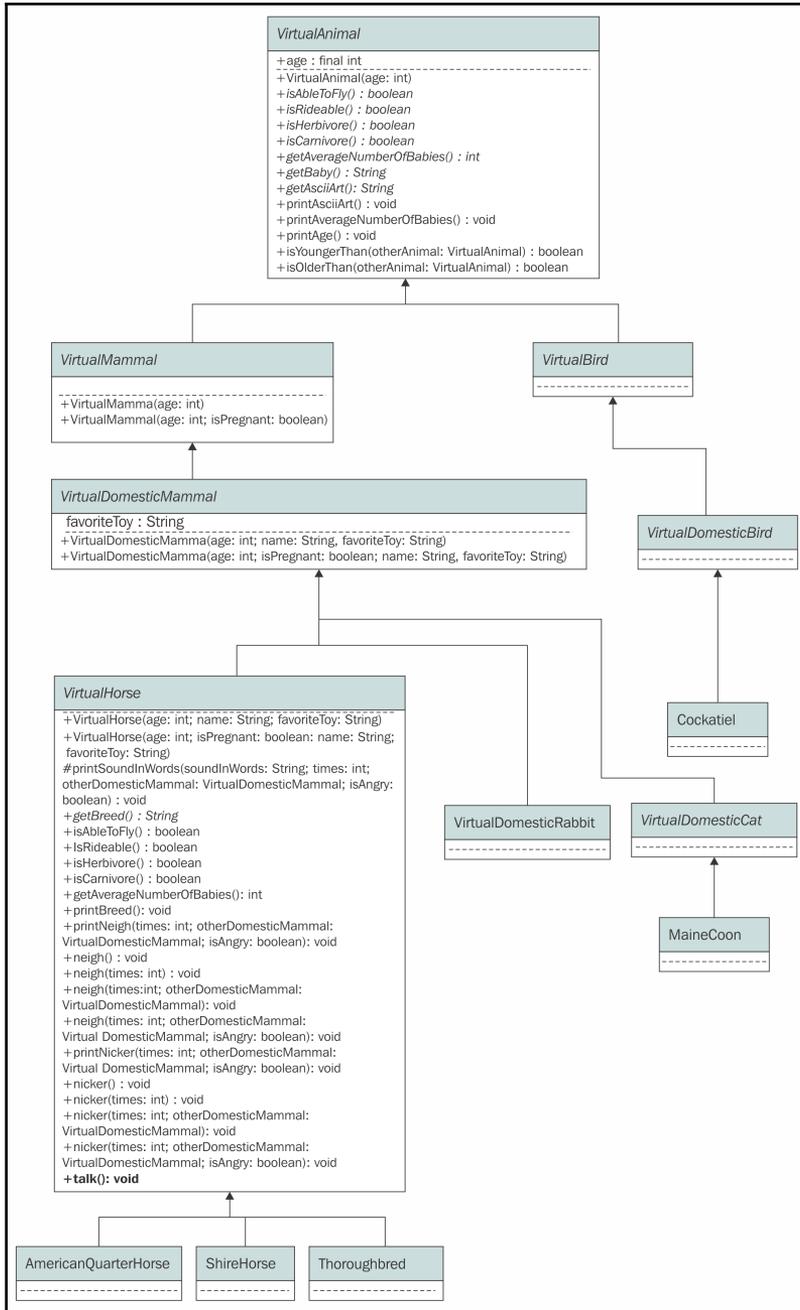
jshell> ImmutableVirtualCreature meowth3 =
...>     meowth2.evolveToVisibilityLevel(25);
meowth3 ==> ImmutableVirtualCreature@78c03f1f
| created variable meowth3 : ImmutableVirtualCreature

jshell> System.out.printf("%d\n", meowth3.visibilityLevel);
25
$7 ==> java.io.PrintStream@52a86356
| created scratch variable $7 : PrintStream

jshell>
```

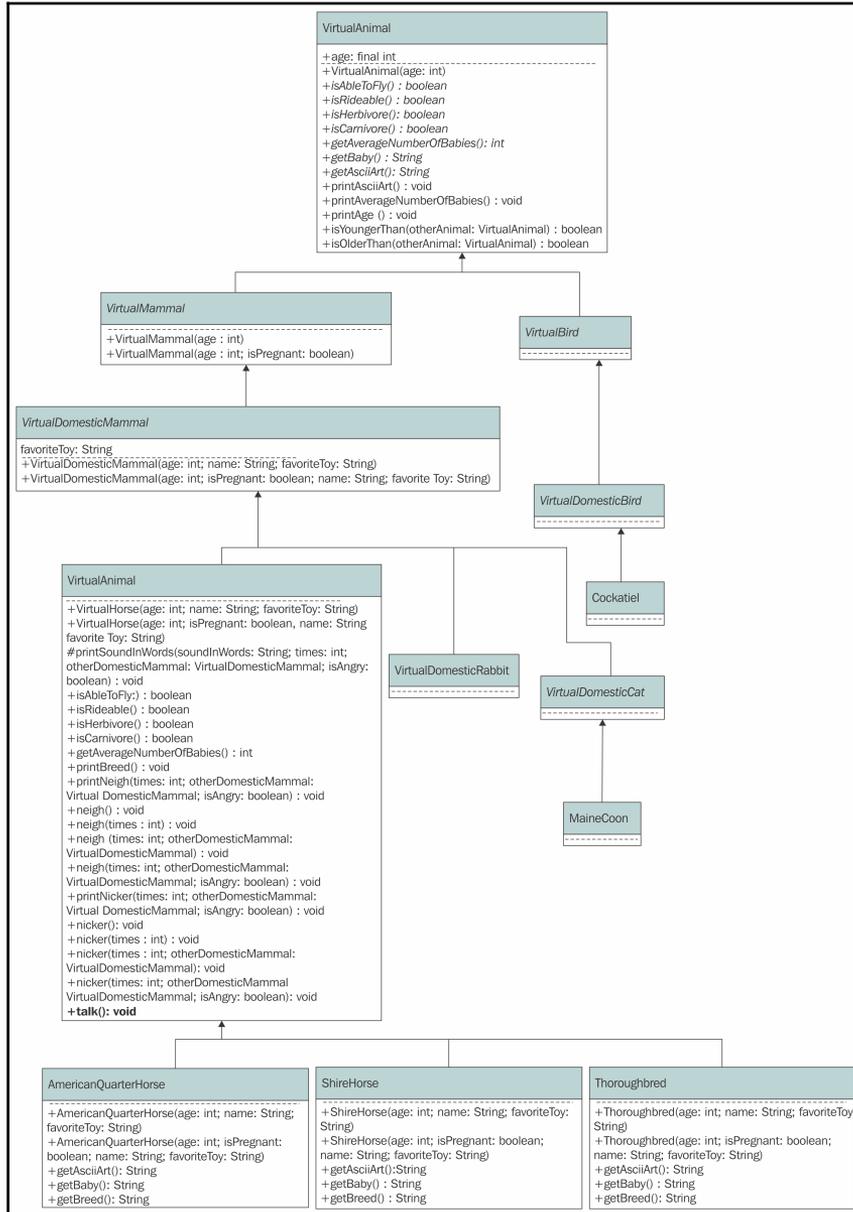
# Chapter 6: Inheritance, Abstraction, Extension, and Specialization







# Chapter 7: Members Inheritance and Polymorphism



```
jshell> AmericanQuarterHorse american =
...>     new AmericanQuarterHorse(
...>         8, "American", "Equi-Spirit Ball");
VirtualAnimal created.
VirtualMammal created.
VirtualDomesticMammal created.
VirtualHorse created.
AmericanQuarterHorse created.
american ==> AmericanQuarterHorse@10bdf5e5
| created variable american : AmericanQuarterHorse

jshell> american.printBreed();
American Quarter Horse

jshell> System.out.println(american instanceof VirtualAnimal);
true

jshell> System.out.println(american instanceof VirtualMammal);
true

jshell> System.out.println(american instanceof VirtualDomesticMammal);
true

jshell> System.out.println(american instanceof VirtualHorse);
true

jshell> System.out.println(american instanceof AmericanQuarterHorse);
true

jshell>
```

```

ShireHorse created.
zelda ==> ShireHorse@31dc339b
| created variable zelda : ShireHorse

jshell> american.printAverageNumberOfBabies();
AQH baby

jshell> american.printAsciiArt();
  >>\.
 /* )`.
 // _)^`^`.
 (, '\ ^-)' ' ' .-.-.-.-. \
 |
 | \ | \
 / \ / \ .-.-.-. '\ ( \ (
 < , ' || \ | ` \ \ -'
 \ \ ( ) | ) /
 |_>|> /_ ] //
 /_ ] /_ ]

jshell> zelda.printAverageNumberOfBabies();
ShireHorse baby

jshell> zelda.printAsciiArt();
          ;;
          ;; '*\
          ;; ' '\
 /' _ '\ . ~ . ~ ' \ / '\ .)
 ;;( ) / |
 ;; ' \ / - . , , ( )
 ) / | ) / |
 || ( \ || ( \
 ( \ ( \

jshell>

```



```
jshell> willow.neigh();
Willow: Neigh

jshell> willow.neigh(2);
Willow: Neigh Neigh

jshell> willow.neigh(2, american);
Willow to American : Neigh Neigh

jshell> willow.neigh(3, zelda, true);
Willow to Zelda : Angry Neigh Neigh Neigh

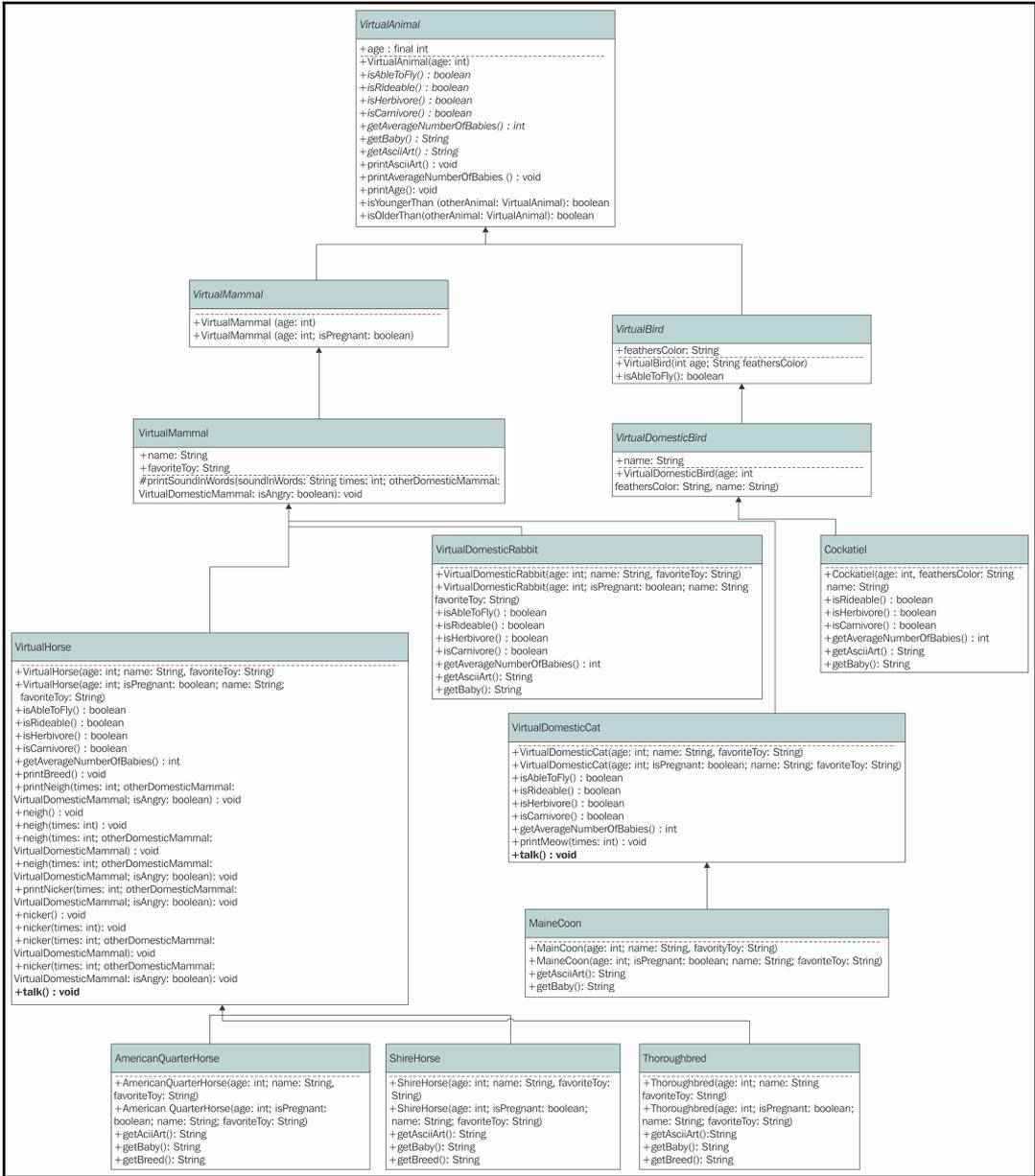
jshell> american.nicker();
American: Nicker

jshell> american.nicker(2);
American: Nicker Nicker

jshell> american.nicker(2, willow);
American to Willow : Nicker Nicker

jshell> american.nicker(3, willow, true);
American to Willow : Angry Nicker Nicker Nicker

jshell>
```



```
jshell> Cockatiel tweety =
  ...>     new Cockatiel(3, "White", "Tweety");
VirtualAnimal created.
VirtualBird created.
VirtualDomesticBird created.
Cockatiel created.
tweety ==> Cockatiel@76707e36
| created variable tweety : Cockatiel

jshell> VirtualDomesticRabbit bunny =
  ...>     new VirtualDomesticRabbit(2, "Bunny", "Sneakers");
VirtualAnimal created.
VirtualMammal created.
VirtualDomesticMammal created.
VirtualDomesticRabbit created.
bunny ==> VirtualDomesticRabbit@1f554b06
| created variable bunny : VirtualDomesticRabbit

jshell> MaineCoon garfield =
  ...>     new MaineCoon(3, "Garfield", "Lassagna");
VirtualAnimal created.
VirtualMammal created.
VirtualDomesticMammal created.
VirtualDomesticCat created.
MaineCoon created.
garfield ==> MaineCoon@1c3a4799
| created variable garfield : MaineCoon

jshell>
```



```
jshell> printf(garfield.name);
Garfield
jshell> printBabies(garfield);
Maine Coon baby Maine Coon baby
jshell> printAsciiArt(garfield);
  ^ ^
 (*~*)
  | |
 /  \

jshell>
```

```
jshell> void makeItTalk(VirtualDomesticMammal domestic) {
    ...>     domestic.talk();
    ...> }
| created method makeItTalk(VirtualDomesticMammal)

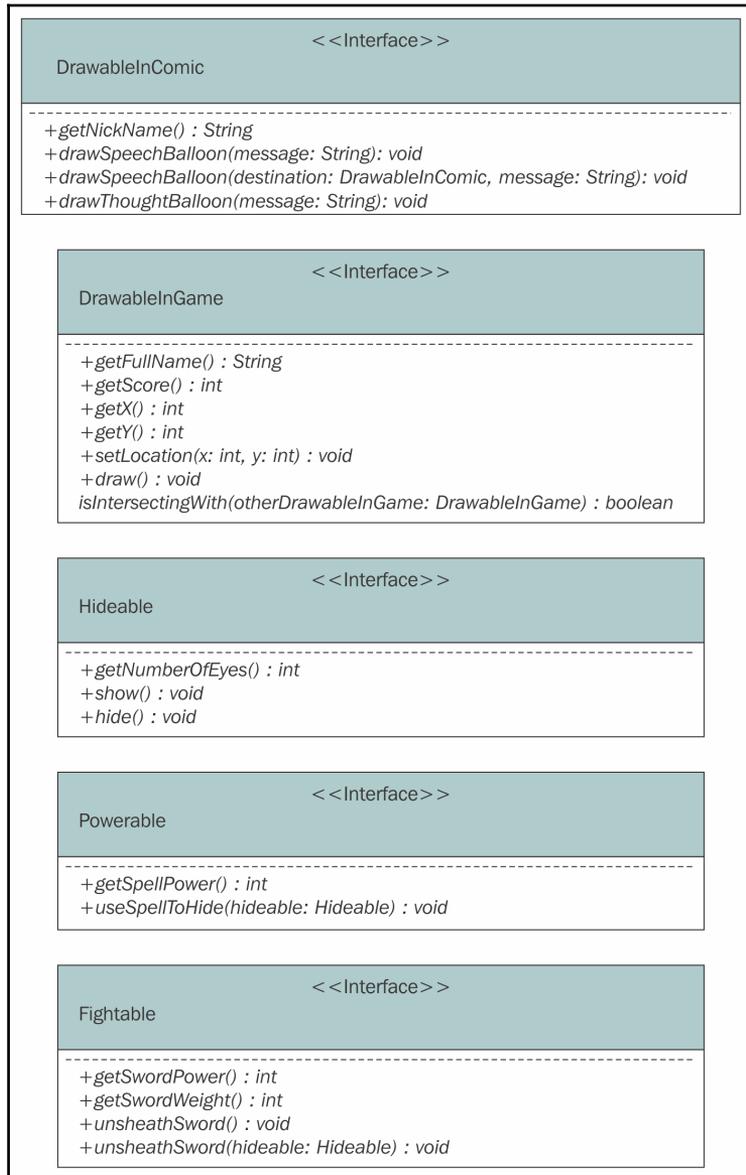
jshell> makeItTalk(bunny);
Bunny: says something

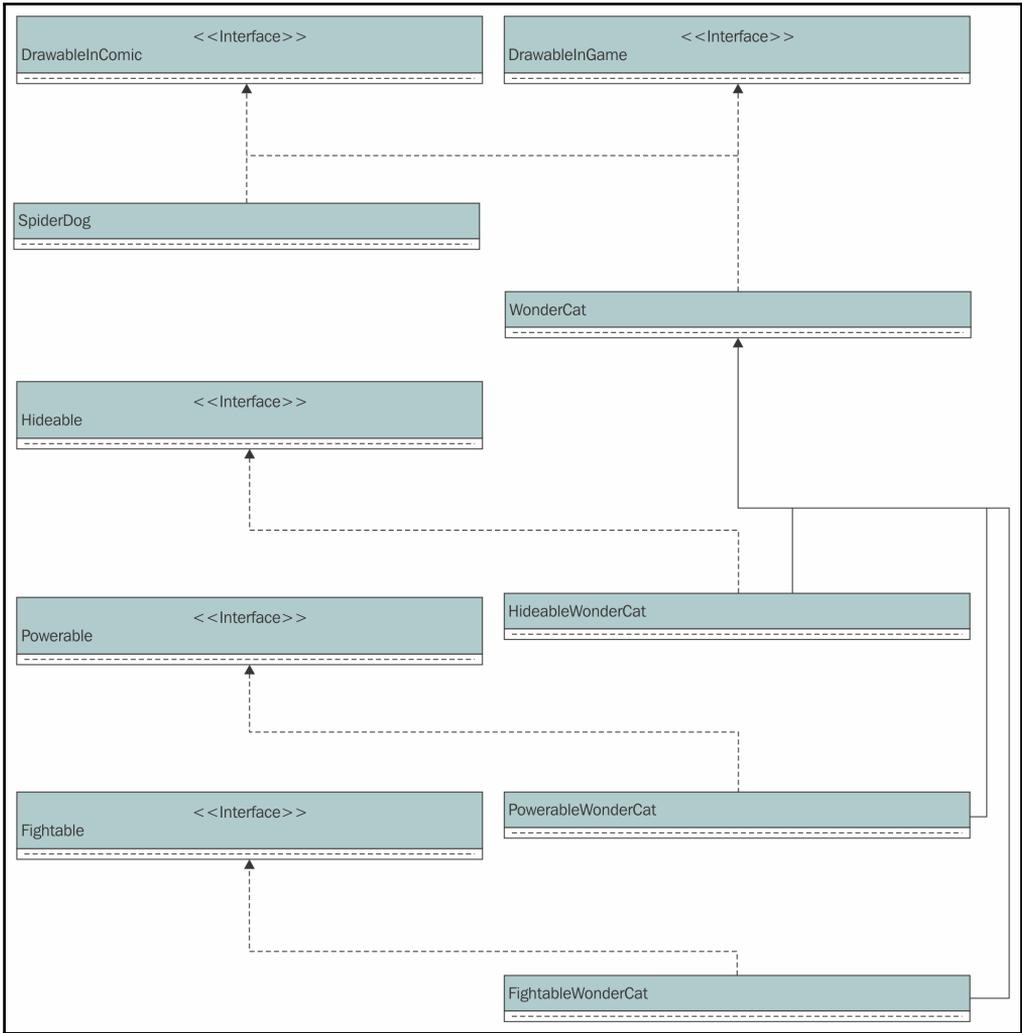
jshell> makeItTalk(garfield);
Garfield: Meow

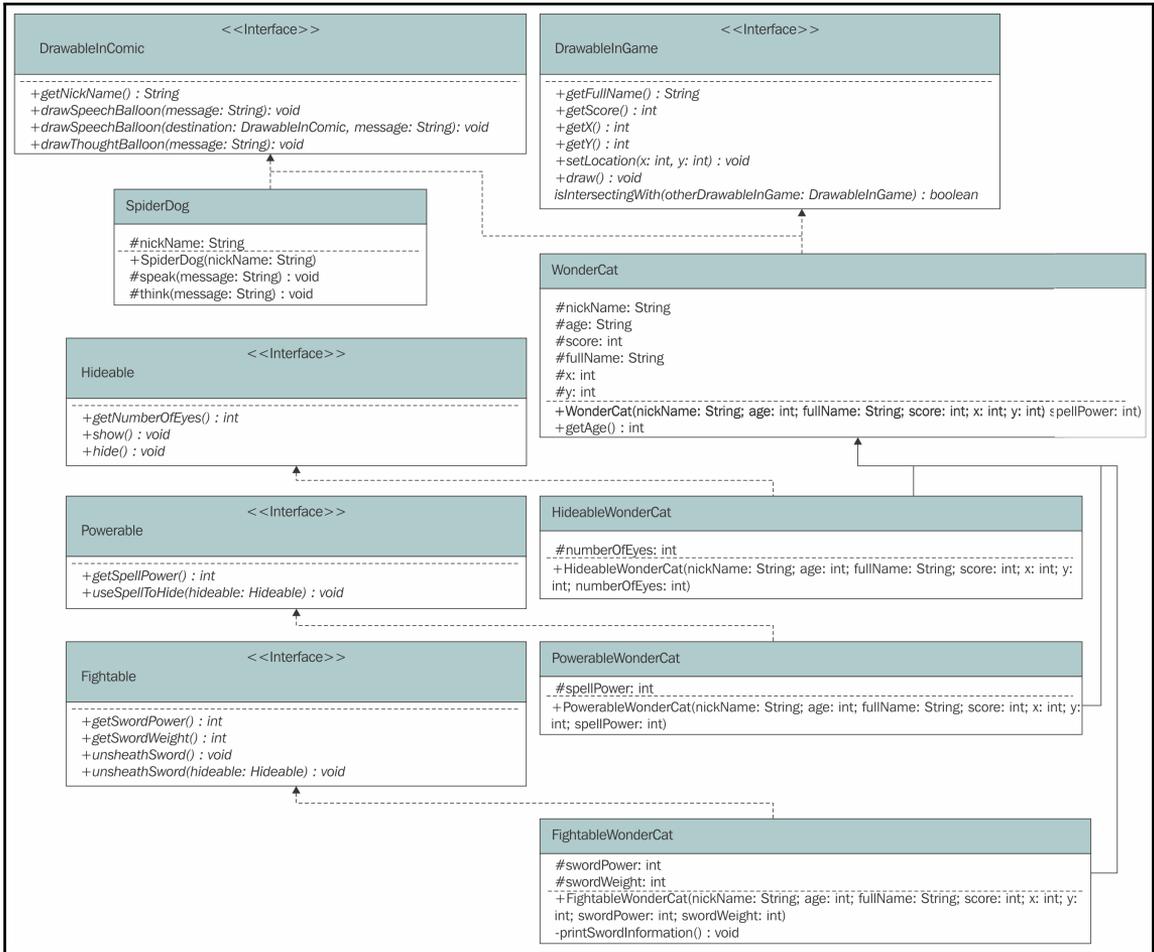
jshell>
```

```
jshell> tweety.printAge();  
I am 3 years old  
  
jshell> bunny.printAge();  
I am 2 years old  
  
jshell> garfield.printAge();  
I am 3 years old  
  
jshell> tweety.isOlderThan(bunny);  
$58 ==> true  
| created scratch variable $58 : boolean  
  
jshell> garfield.isYoungerThan(tweety);  
$59 ==> false  
| created scratch variable $59 : boolean  
  
jshell> bunny.isYoungerThan(garfield);  
$60 ==> true  
| created scratch variable $60 : boolean  
  
jshell>
```

# Chapter 8: Contract Programming with Interfaces







```
jshell> spiderDog1 instanceof SpiderDog
$17 ==> true
| created scratch variable $17 : boolean

jshell> spiderDog1 instanceof DrawableInComic
$18 ==> true
| created scratch variable $18 : boolean

jshell> wonderCat1 instanceof WonderCat
$19 ==> true
| created scratch variable $19 : boolean

jshell> wonderCat1 instanceof DrawableInComic
$20 ==> true
| created scratch variable $20 : boolean

jshell> wonderCat1 instanceof DrawableInGame
$21 ==> true
| created scratch variable $21 : boolean

jshell> hideableWonderCat1 instanceof WonderCat
$22 ==> true
| created scratch variable $22 : boolean

jshell> hideableWonderCat1 instanceof HideableWonderCat
$23 ==> true
| created scratch variable $23 : boolean

jshell> hideableWonderCat1 instanceof DrawableInComic
$24 ==> true
| created scratch variable $24 : boolean

jshell> hideableWonderCat1 instanceof DrawableInGame
$25 ==> true
| created scratch variable $25 : boolean

jshell> hideableWonderCat1 instanceof Hideable
$26 ==> true
| created scratch variable $26 : boolean
```

```
jshell> powerableWonderCat1 instanceof WonderCat
$27 ==> true
| created scratch variable $27 : boolean

jshell> powerableWonderCat1 instanceof PowerableWonderCat
$28 ==> true
| created scratch variable $28 : boolean

jshell> powerableWonderCat1 instanceof DrawableInComic
$29 ==> true
| created scratch variable $29 : boolean

jshell> powerableWonderCat1 instanceof DrawableInGame
$30 ==> true
| created scratch variable $30 : boolean

jshell> powerableWonderCat1 instanceof Powerable
$31 ==> true
| created scratch variable $31 : boolean

jshell> fightableWonderCat1 instanceof WonderCat
$32 ==> true
| created scratch variable $32 : boolean

jshell> fightableWonderCat1 instanceof FightableWonderCat
$33 ==> true
| created scratch variable $33 : boolean

jshell> fightableWonderCat1 instanceof DrawableInComic
$34 ==> true
| created scratch variable $34 : boolean

jshell> fightableWonderCat1 instanceof DrawableInGame
$35 ==> true
| created scratch variable $35 : boolean

jshell> fightableWonderCat1 instanceof Fightable
$36 ==> true
| created scratch variable $36 : boolean
```

## Chapter 9: Advanced Contract Programming with Interfaces

```
jshell> /types
|   interface DrawableInComic
|   interface DrawableInGame
|   interface Hideable
|   interface Powerable
|   interface Fightable
|   class SpiderDog
|   class WonderCat
|   class HideableWonderCat
|   class PowerableWonderCat
|   class FightableWonderCat
jshell>
```

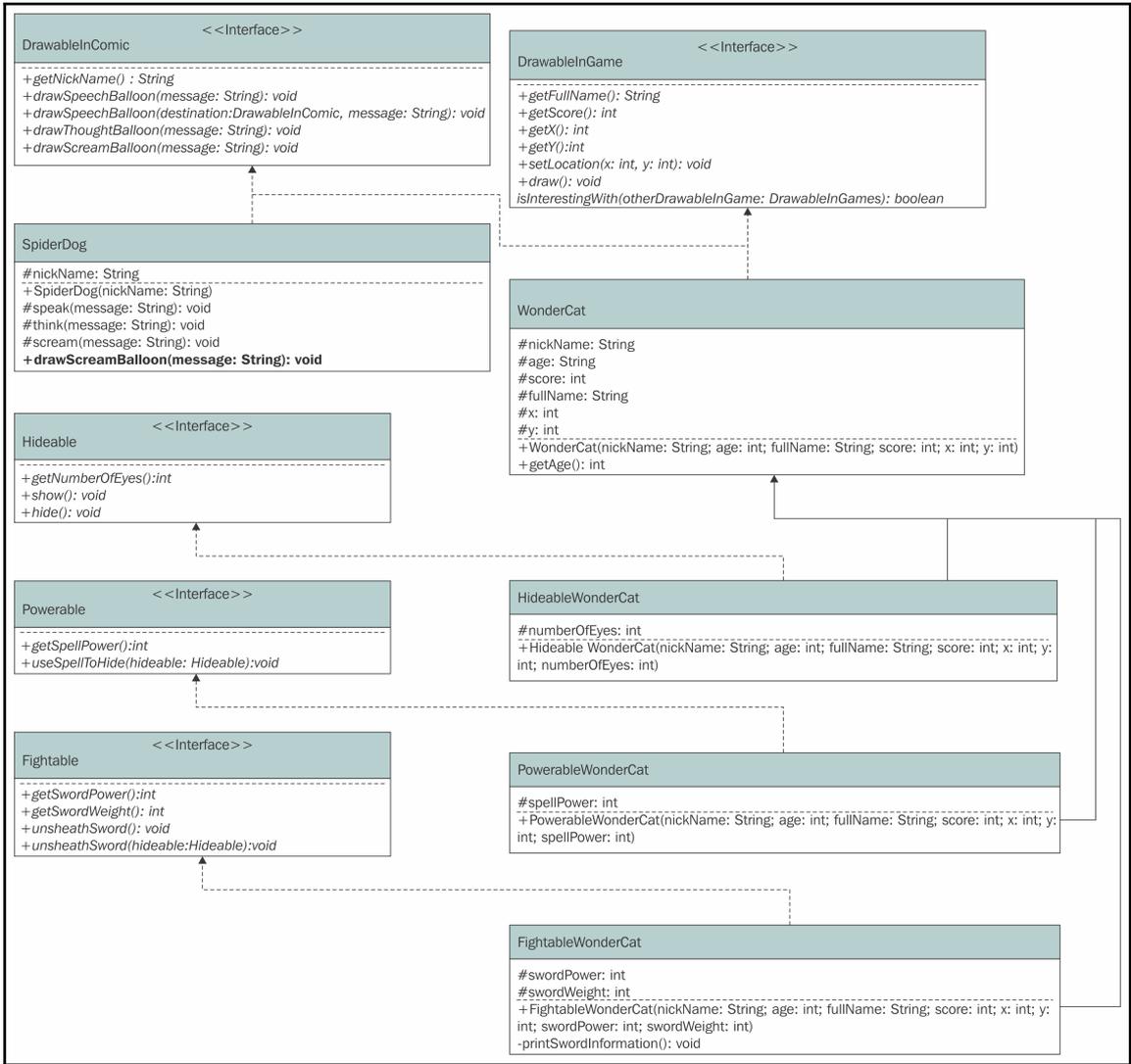
```
jshell> doSomethingWithWonderCat(misterHideable);  
My name is Mr. John Hideable and you can see my 3 eyes.
```

```
jshell> doSomethingWithWonderCat(spartan);  
Sir Spartan unsheaths his sword.  
Sword power: 100. Sword weight: 50.
```

```
jshell> doSomethingWithWonderCat(merlin);  
Spell power: 200
```

```
jshell> doSomethingWithWonderCat(oliver);  
This WonderCat isn't cool.
```

```
jshell>
```



```
| update replaced variable spiderDog1, reset to null
| update replaced variable teddy, reset to null
| update replaced variable winston, reset to null
| update overwrote class SpiderDog

jshell> SpiderDog rocky = new SpiderDog("Rocky");
rocky ==> SpiderDog@2f943d71
| created variable rocky : SpiderDog

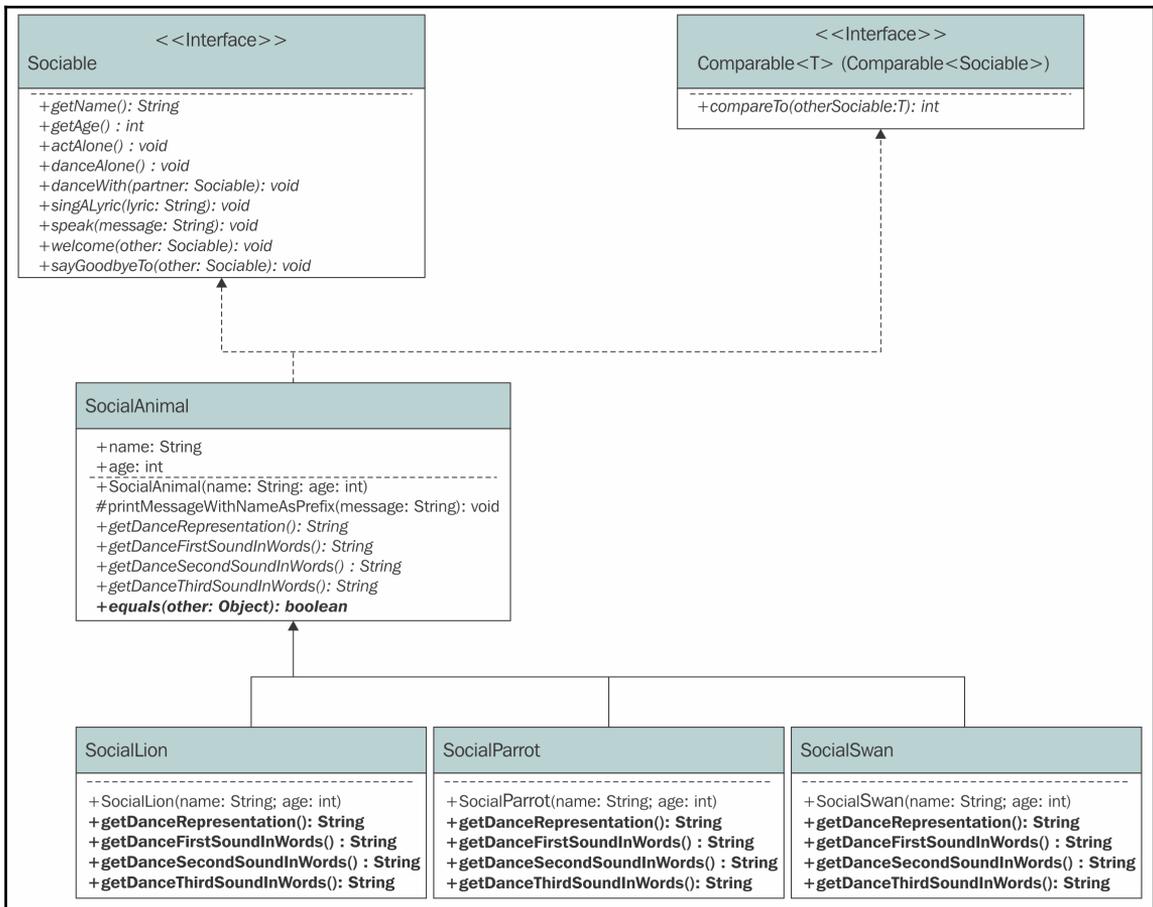
jshell> FightableWonderCat maggie =
...>     new FightableWonderCat("Maggie", 2,
...>     "Mrs. Maggie", 5000000, 10, 10, 80, 30);
maggie ==> FightableWonderCat@4b553d26
| created variable maggie : FightableWonderCat

jshell> rocky.drawScreamBalloon("I am Rocky!");
Rocky screams +++ I am Rocky! +++

jshell> maggie.drawScreamBalloon("I am Mrs. Maggie!");
Maggie -> Meeoow Meeoow

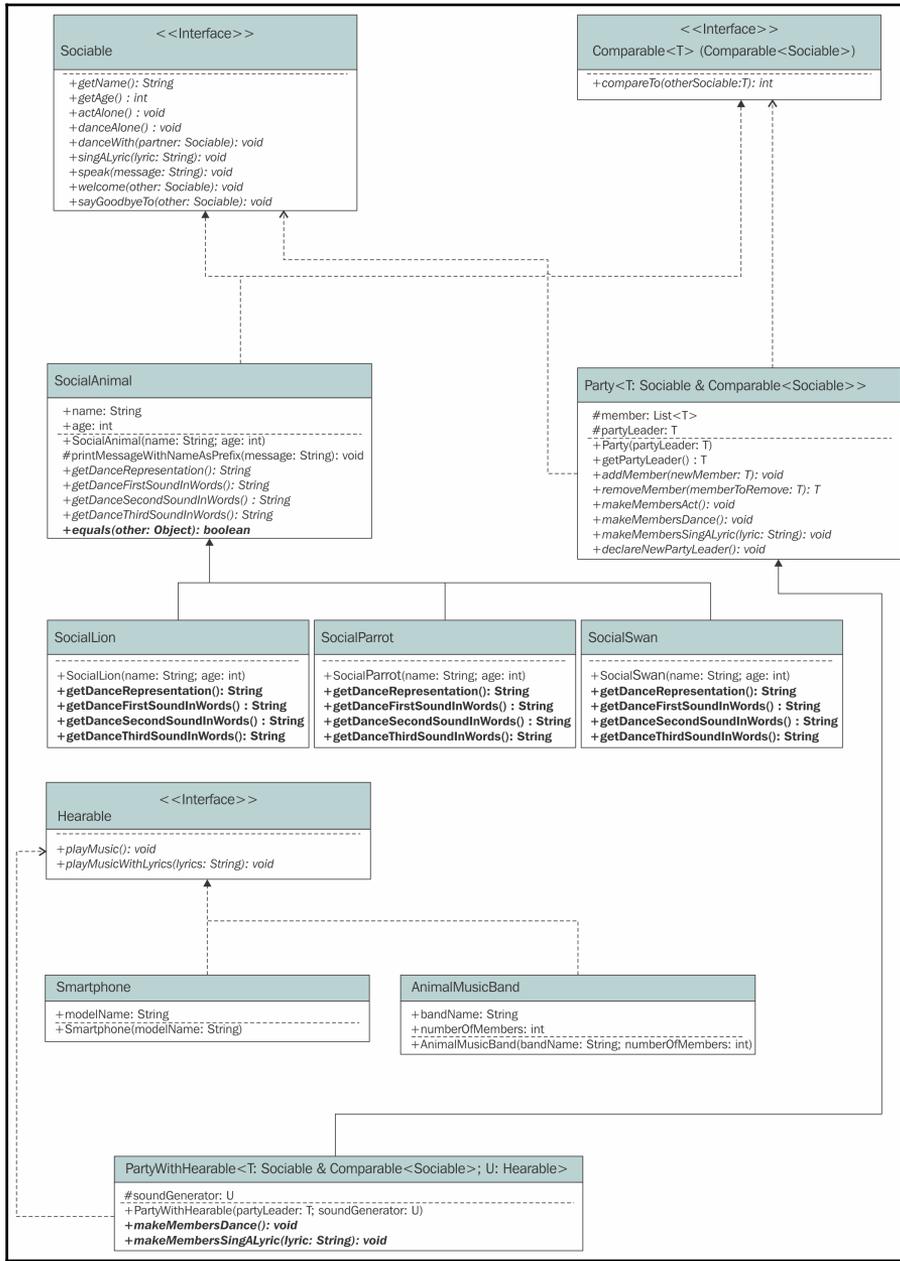
jshell>
```

# Chapter 10: Maximization of Code Reuse with Generics



```
jshell> lionsParty.addMember(rio);
| Error:
| incompatible types: SocialParrot cannot be converted to SocialLion
| lionsParty.addMember(rio);
|                        ^-^
|
jshell> parrotsParty.addMember(simba);
| Error:
| incompatible types: SocialLion cannot be converted to SocialParrot
| parrotsParty.addMember(simba);
|                        ^---^
|
jshell>
```

# Chapter 11: Advanced Generics



```
jshell> Smartphone android = new Smartphone("Super Android Smartphone");
android ==> Smartphone@184f6be2
| created variable android : Smartphone

jshell> PartyWithHearable<SocialLion, Smartphone> nalaParty =
...> new PartyWithHearable<>(nala, android);
nalaParty ==> PartyWithHearable@1f7030a6
| created variable nalaParty : PartyWithHearable<SocialLion, Smartphone>

jshell>

jshell> nalaParty.addMember(simba);
Nala welcomes Simba

jshell> nalaParty.addMember(mufasa);
Nala welcomes Mufasa

jshell> nalaParty.addMember(scar);
Nala welcomes Scar

jshell>
```

```

jshell> nalaParty.makeMembersDance();
Super Android Smartphone starts playing music.
cha-cha-cha untz untz untz
Nala dances alone *-* ^\/^ (-)
Simba dances alone *-* ^\/^ (-)
Mufasa dances alone *-* ^\/^ (-)
Scar dances alone *-* ^\/^ (-)

jshell> try {
...>     nalaParty.removeMember(mufasa);
...> } catch (CannotRemovePartyLeaderException e) {
...>     System.out.println(
...>         "We cannot remove the party leader.");
...> }
Mufasa says goodbye to Nala RoarRrooaarrRrrrrrrroooooaaarrrr

jshell> try {
...>     nalaParty.declareNewPartyLeader();
...> } catch (InsufficientMembersException e) {
...>     System.out.println(
...>         String.format("We just have %s member",
...>             e.getNumberOfMembers()));
...> }
Nala says: Scar is our new party leader. *-* ^\/^ (-)
Scar dances with Nala *-* ^\/^ (-)

jshell> nalaParty.makeMembersSingALyric("It's the eye of the tiger");
Super Android Smartphone starts playing music with lyrics.
untz untz untz It's the eye of the tiger untz untz
Nala sings It's the eye of the tiger Roar Rrooaarr Rrrrrrrroooooaaarrrr
Simba sings It's the eye of the tiger Roar Rrooaarr Rrrrrrrroooooaaarrrr
Scar sings It's the eye of the tiger Roar Rrooaarr Rrrrrrrroooooaaarrrr

jshell>

```

```
jshell> AnimalMusicBand band = new AnimalMusicBand(
...>     "Black Eyed Paws", 4);
band ==> AnimalMusicBand@3b088d51
| created variable band : AnimalMusicBand

jshell> PartyWithHearable<SocialParrot, AnimalMusicBand> ramboParty =
...>     new PartyWithHearable<>(rambo, band);
ramboParty ==> PartyWithHearable@74650e52
| created variable ramboParty : PartyWithHearable<SocialParrot, AnimalMusicBand>

jshell>

jshell> ramboParty.addMember(rio);
Rambo welcomes Rio

jshell> ramboParty.addMember(woody);
Rambo welcomes Woody

jshell> ramboParty.addMember(thor);
Rambo welcomes Thor

jshell>
```

```
jshell> ramboParty.makeMembersDance();
Our name is Black Eyed Paws. We are 4.
Meow Meow Woof Woof Meow Meow
Rambo dances alone /|\ --- %% ==+
Rio dances alone /|\ --- %% ==+
Woody dances alone /|\ --- %% ==+
Thor dances alone /|\ --- %% ==+

jshell> try {
...>     ramboParty.removeMember(rio);
...> } catch (CannotRemovePartyLeaderException e) {
...>     System.out.println(
...>         "We cannot remove the party leader.");
...> }
Rio says goodbye to Rambo YeahYeeeahYeeeeeaaah

jshell> try {
...>     ramboParty.declareNewPartyLeader();
...> } catch (InsufficientMembersException e) {
...>     System.out.println(
...>         String.format("We just have %s member",
...>             e.getNumberOfMembers()));
...> }
Rambo says: Thor is our new party leader. /|\ --- %% ==+
Thor dances with Rambo /|\ --- %% ==+

jshell> ramboParty.makeMembersSingALyric("Turn up the radio");
Black Eyed Paws asks you to sing together.
Meow Woof Turn up the radio Woof Meow
Rambo sings Turn up the radio Yeah Yeeeah Yeeeeeaaah
Woody sings Turn up the radio Yeah Yeeeah Yeeeeeaaah
Thor sings Turn up the radio Yeah Yeeeah Yeeeeeaaah

jshell>
```

## Chapter 12: Object-Oriented, Functional Programming, and Lambda Expressions

```
jshell>
jshell> import java.util.stream.Collectors;
jshell> import java.util.stream.IntStream;
jshell>
jshell> List<Integer> range1to20 =
...>     IntStream.rangeClosed(1, 20).boxed().collect(Collectors.toList());
range1to20 ==> [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]
| created variable range1to20 : List<Integer>
jshell>
```

```
jshell> TestDivisibleBy5 testDivisibleBy5 = new TestDivisibleBy5();
testDivisibleBy5 ==> TestDivisibleBy5@6b09bb57
| created variable testDivisibleBy5 : TestDivisibleBy5

jshell> List<Integer> divisibleBy5Numbers =
...> filterNumbersWithTestable(range1to20, testDivisibleBy5);
divisibleBy5Numbers ==> [5, 10, 15, 20]
| created variable divisibleBy5Numbers : List<Integer>

jshell> System.out.println(divisibleBy5Numbers);
[5, 10, 15, 20]

jshell>

jshell> TestGreaterThan10 testGreaterThan10 = new TestGreaterThan10();
testGreaterThan10 ==> TestGreaterThan10@eec5a4a
| created variable testGreaterThan10 : TestGreaterThan10

jshell> List<Integer> greaterThan10Numbers =
...> filterNumbersWithTestable(range1to20, testGreaterThan10);
greaterThan10Numbers ==> [11, 12, 13, 14, 15, 16, 17, 18, 19, 20]
| created variable greaterThan10Numbers : List<Integer>

jshell> System.out.println(greaterThan10Numbers);
[11, 12, 13, 14, 15, 16, 17, 18, 19, 20]

jshell>
```

```
jshell> IntPredicate divisibleBy5 = n -> n % 5 == 0;
divisibleBy5 ==> $Lambda$13/1262822392@731a74c
| created variable divisibleBy5 : IntPredicate

jshell> List<Integer> divisibleBy5Numbers2 =
...> filterNumbersWithPredicate(range1to20, divisibleBy5);
divisibleBy5Numbers2 ==> [5, 10, 15, 20]
| created variable divisibleBy5Numbers2 : List<Integer>

jshell> System.out.println(divisibleBy5Numbers2);
[5, 10, 15, 20]

jshell> List<Integer> greaterThan10Numbers2 =
...> filterNumbersWithPredicate(range1to20, n -> n > 10);
greaterThan10Numbers2 ==> [11, 12, 13, 14, 15, 16, 17, 18, 19, 20]
| created variable greaterThan10Numbers2 : List<Integer>

jshell> System.out.println(greaterThan10Numbers);
[11, 12, 13, 14, 15, 16, 17, 18, 19, 20]

jshell>
```

```
jshell> List<Integer> divisibleBy3Numbers2 =
...>     range1to20.stream().filter(
...>         n -> n % 3 == 0).collect(
...>         Collectors.toList());
divisibleBy3Numbers2 ==> [3, 6, 9, 12, 15, 18]
| created variable divisibleBy3Numbers2 : List<Integer>

jshell> divisibleBy3Numbers2.forEach(n -> System.out.println(n));
3
6
9
12
15
18

jshell>

jshell> divisibleBy3Numbers2.forEach(System.out::println);
3
6
9
12
15
18

jshell>
```

```
jshell> List<Integer> divisibleBy3Numbers2 =
...>     range1to20.stream().filter(
...>         n -> n % 3 == 0).collect(
...>         Collectors.toList());
divisibleBy3Numbers2 ==> [3, 6, 9, 12, 15, 18]
|   created variable divisibleBy3Numbers2 : List<Integer>

jshell> divisibleBy3Numbers2.forEach(n -> System.out.println(n));
3
6
9
12
15
18

jshell>

jshell> divisibleBy3Numbers2.forEach(System.out::println);
3
6
9
12
15
18

jshell>
```

```
| update replaced variable repository, reset to null
| update overwrote class MemoryMobileGameRepository

jshell>

jshell> MemoryMobileGameRepository repository = new MemoryMobileGameRepository()
repository ==> MemoryMobileGameRepository@7a765367
| modified variable repository : MemoryMobileGameRepository
| update overwrote variable repository : MemoryMobileGameRepository

jshell> Optional<MobileGame> optionalMobileGame1 =
...> repository.getByPlayersCountAndHighestScore(750000, 152000);
optionalMobileGame1 ==> Optional[Id: 4; Name: Mario vs Kong III; Highest score: 152000; Lowest score ...
| created variable optionalMobileGame1 : Optional<MobileGame>

jshell> if (optionalMobileGame1.isPresent()) {
...> MobileGame mobileGame1 = optionalMobileGame1.get();
...> System.out.println(mobileGame1);
...> } else {
; ...> System.out.println("No mobile game matches the specified criteria.")
...> }
Id: 4; Name: Mario vs Kong III; Highest score: 152000; Lowest score: 1500; Players count: 750000

jshell> Optional<MobileGame> optionalMobileGame2 =
...> repository.getByPlayersCountAndHighestScore(670000, 829340);
optionalMobileGame2 ==> Optional.empty
| created variable optionalMobileGame2 : Optional<MobileGame>

jshell> if (optionalMobileGame2.isPresent()) {
...> MobileGame mobileGame2 = optionalMobileGame2.get();
...> System.out.println(mobileGame2);
...> } else {
; ...> System.out.println("No mobile game matches the specified criteria.")
...> }
No mobile game matches the specified criteria.

jshell>
```

# Chapter 13: Modularity in Java 9

