



ORACLE

Academy



Oracle Academy Java for AP Computer Science A

2-3

Introduction to Object-Oriented Programming Concepts

ORACLE
Academy



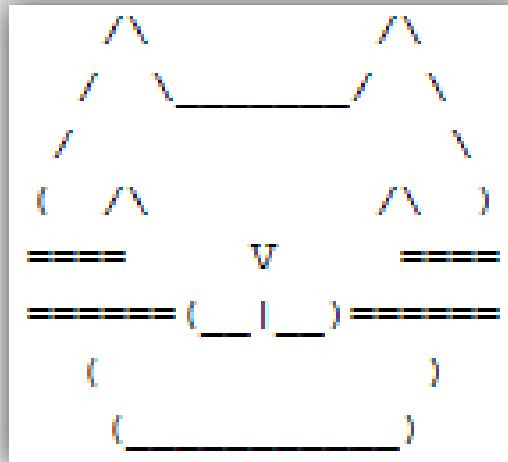
Objectives

- This lesson covers the following objectives:
 - Differentiate between procedural and object-oriented programming
 - Understand a class as a blueprint for an object
 - Understand a class is used to create instances of an object
 - Model objects as a combination of ...
 - Properties (data fields)
 - Behaviors (methods)



Review

- So far, we've taken ...
 - Decades of computer science innovation
 - Gigabytes of modern computing power
- And much like the Internet ...
 - We've made a cat!

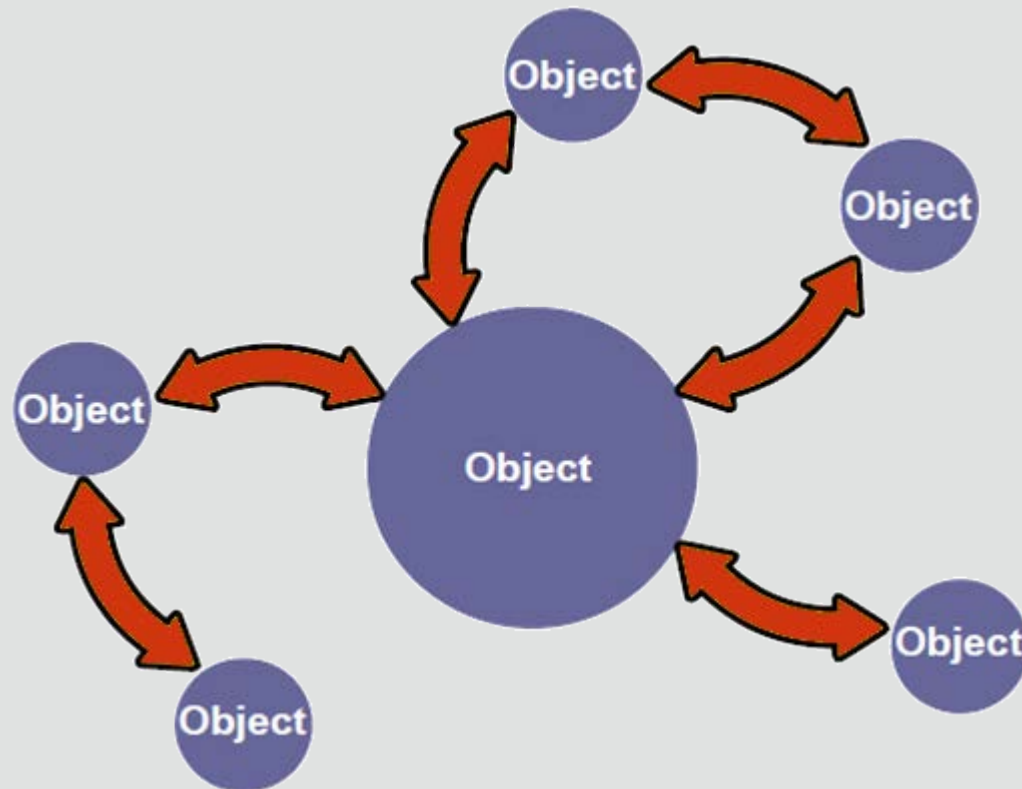


Java Can Do More!

- Procedural languages ...
 - Read one line at a time
 - The C language is procedural
- Object-oriented languages...
 - Read one line at a time
 - Model objects through code
 - Emphasize object interaction
 - Allow interaction without a prescribed order
 - Java and C++ are object-oriented languages

Object-Oriented Programming

- Interaction of objects
- No prescribed sequence



Exercise 1



- Go to <https://objectstorage.ca-toronto-1.oraclecloud.com/n/yzr73ksbiwdp/b/Games/o/JavaPuzzleBall/index.html>
- Play Basic Puzzles 1 through 5
 - Your Goal: Design a solution that deflects the ball to Duke
- Consider the following:
 - What objects do you find on the field of play?
 - What happens when you put a triangle wall or simple wall icon on the blue wheel?





About Java Puzzle Ball

- Play a set of puzzles
- Become familiar with the game mechanics
- Consider questions as you play
- Listen to the lesson's debriefing on what you've observed
- Apply your observations to understand Java concepts





Object Types

- What objects did you find on the field of play?

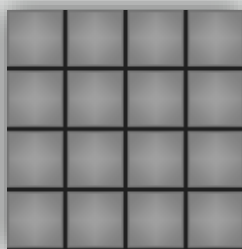
- Ball



- Duke



- LevelGeometry



- RedBumper



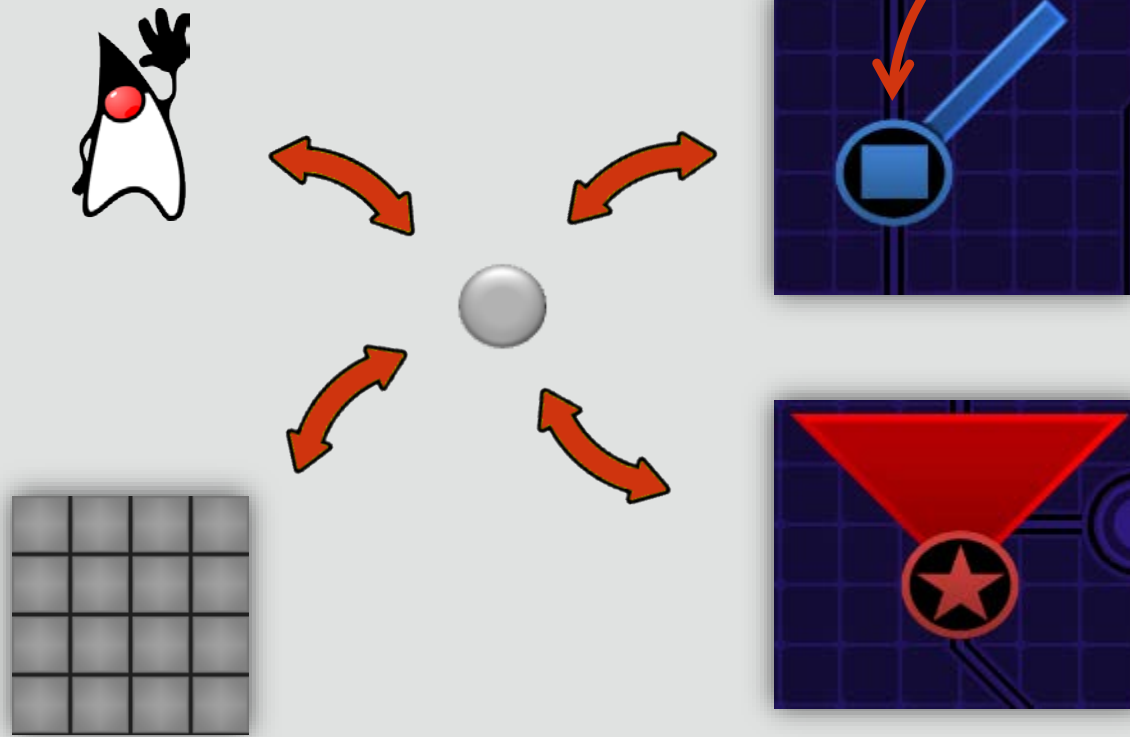
- BlueBumper





Object Interaction

- Interaction of objects
- No prescribed sequence





BlueBumper Objects

- What happens when you put a triangle wall or simple wall icon on a blue wheel?
- A wall appears on every instance of a blue bumper object
- Walls give bumpers behaviors that deflect and interact with the ball
- All blue bumper instances share these same behaviors

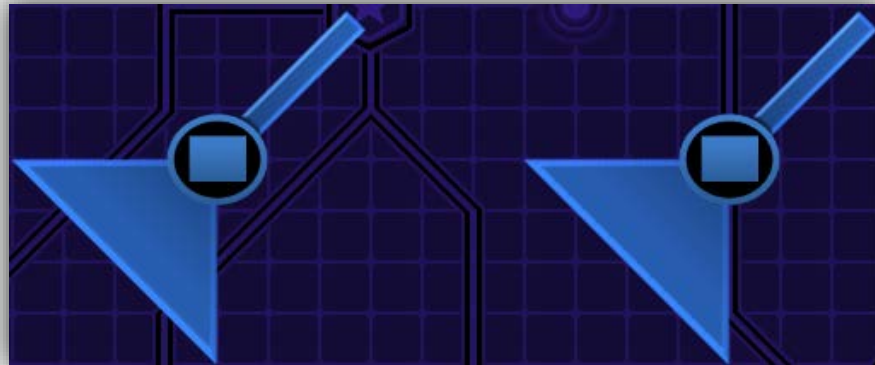




Describing a BlueBumper

- Properties:

- Color
- Shape
- x-position
- y-position



- Behaviors:

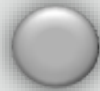
- Make ping sound
- Flash
- Deflect ball (via Simple Wall)
- Deflect ball (via Triangle Wall)



Describing a Ball

- Properties:

- Direction
- x-position
- y-position



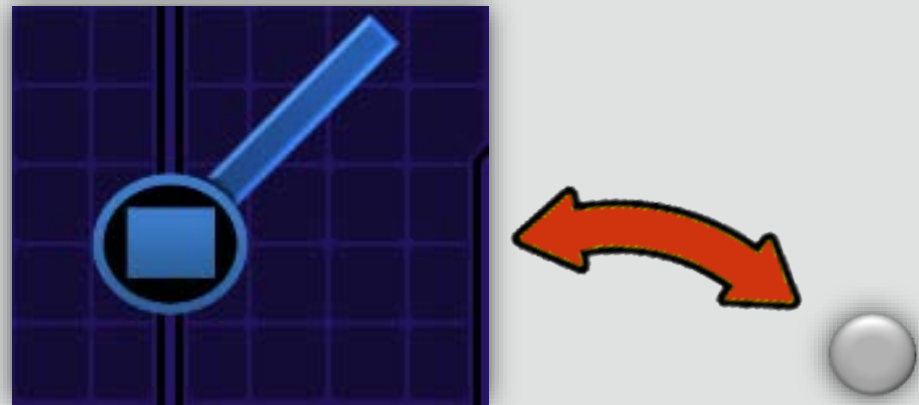
- Behaviors:

- Make ping sound
- Change direction
- Change x-position
- Change y-position



BlueBumper and Ball Interaction

- Interaction occurs when the BlueBumper deflects the Ball. When this happens ...
- The Ball's properties change:
 - The Ball travels in a different direction
 - The Ball's future x-position and y-position change
- The BlueBumper performs behaviors:
 - Makes ping sound
 - Flashes





Why Does This Matter?

- We've observed important aspects of object-oriented programming
- Remember these observations as lessons and exercises become increasingly technical
 - Objects can be described as a combination of properties and behaviors
 - There may be many instances of the same object type
 - All instances of an object share the same behaviors
 - Objects may interact with each other, possibly affecting each other's properties and triggering other behaviors

A Different Example

- Properties:

- Name
- Age
- Breed
- Favorite Food

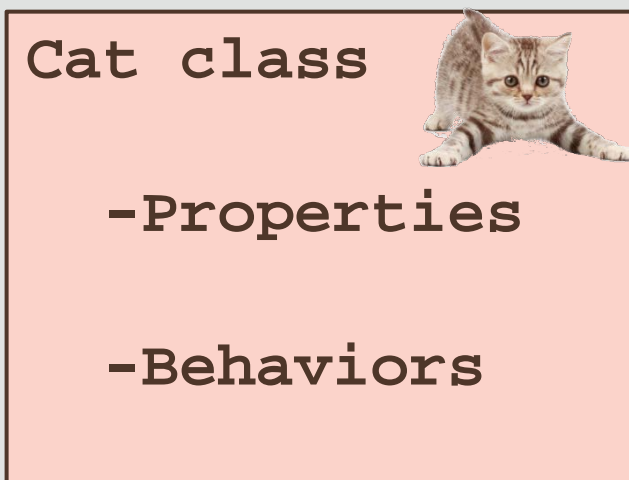


- Behaviors:

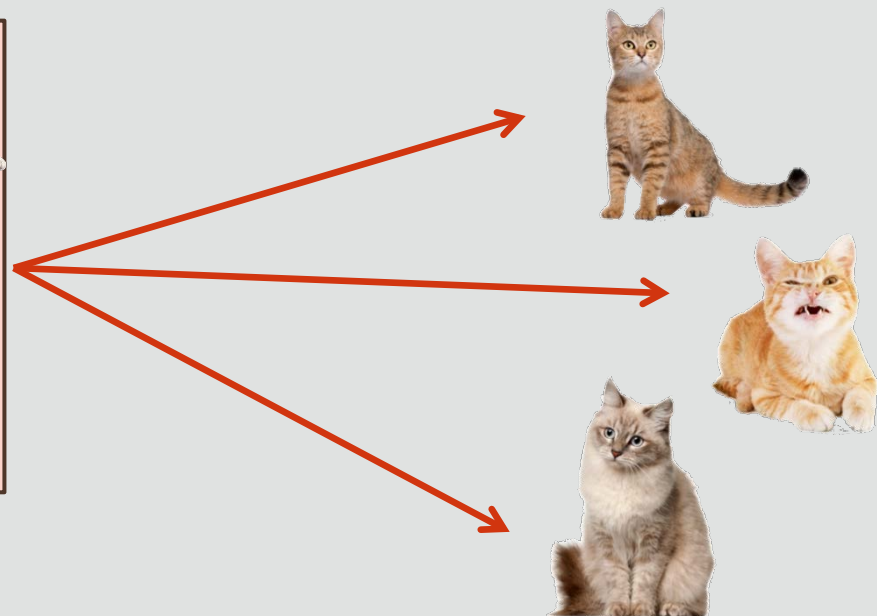
- Make meow sound
- Play
- Wash
- Eat
- Hunt

Classes and Instances

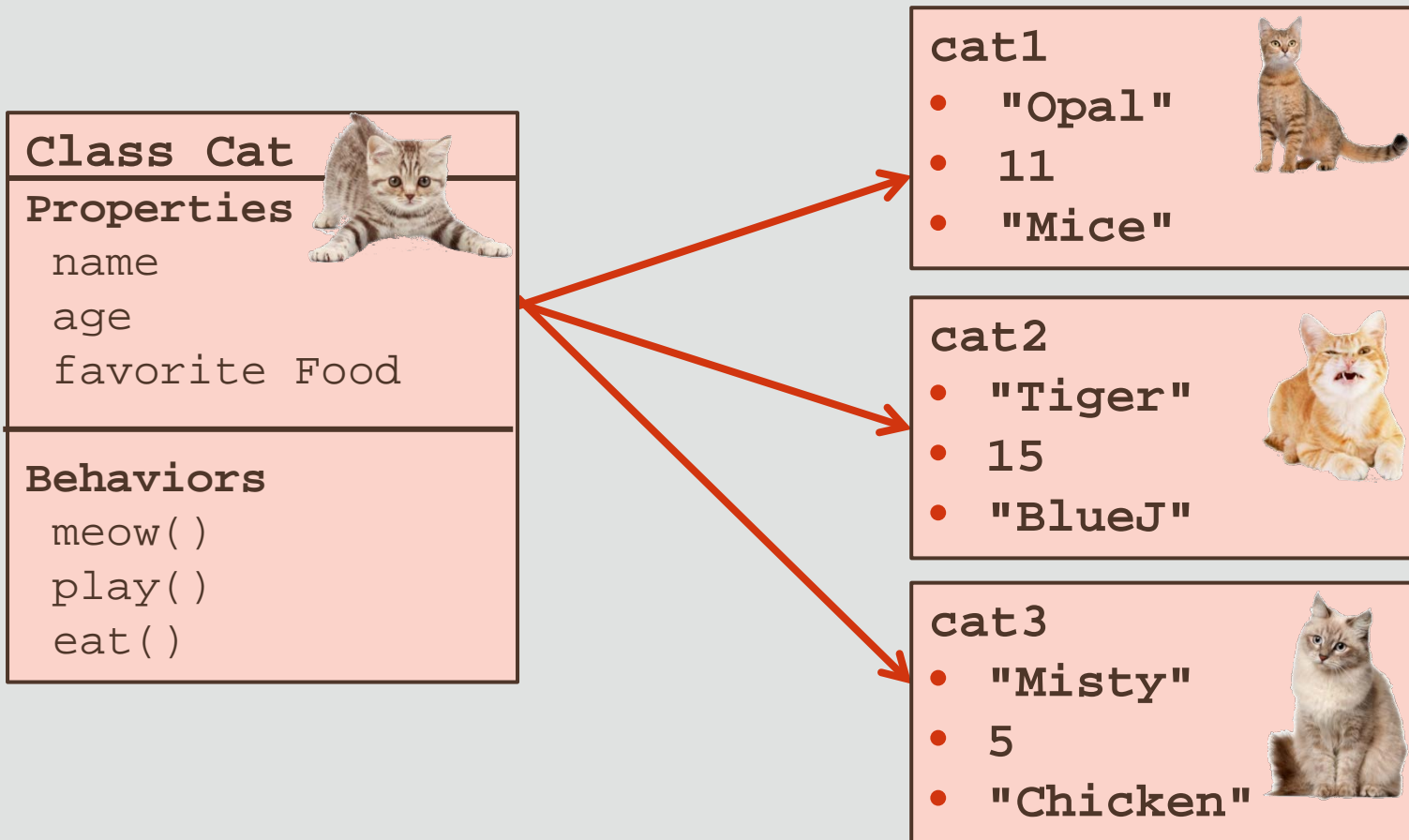
- The combination of properties and behaviors is ...
 - Called a class
 - A blueprint or recipe for an object
 - Used to create object instances



Object instances



Creating New Instances from a Blueprint



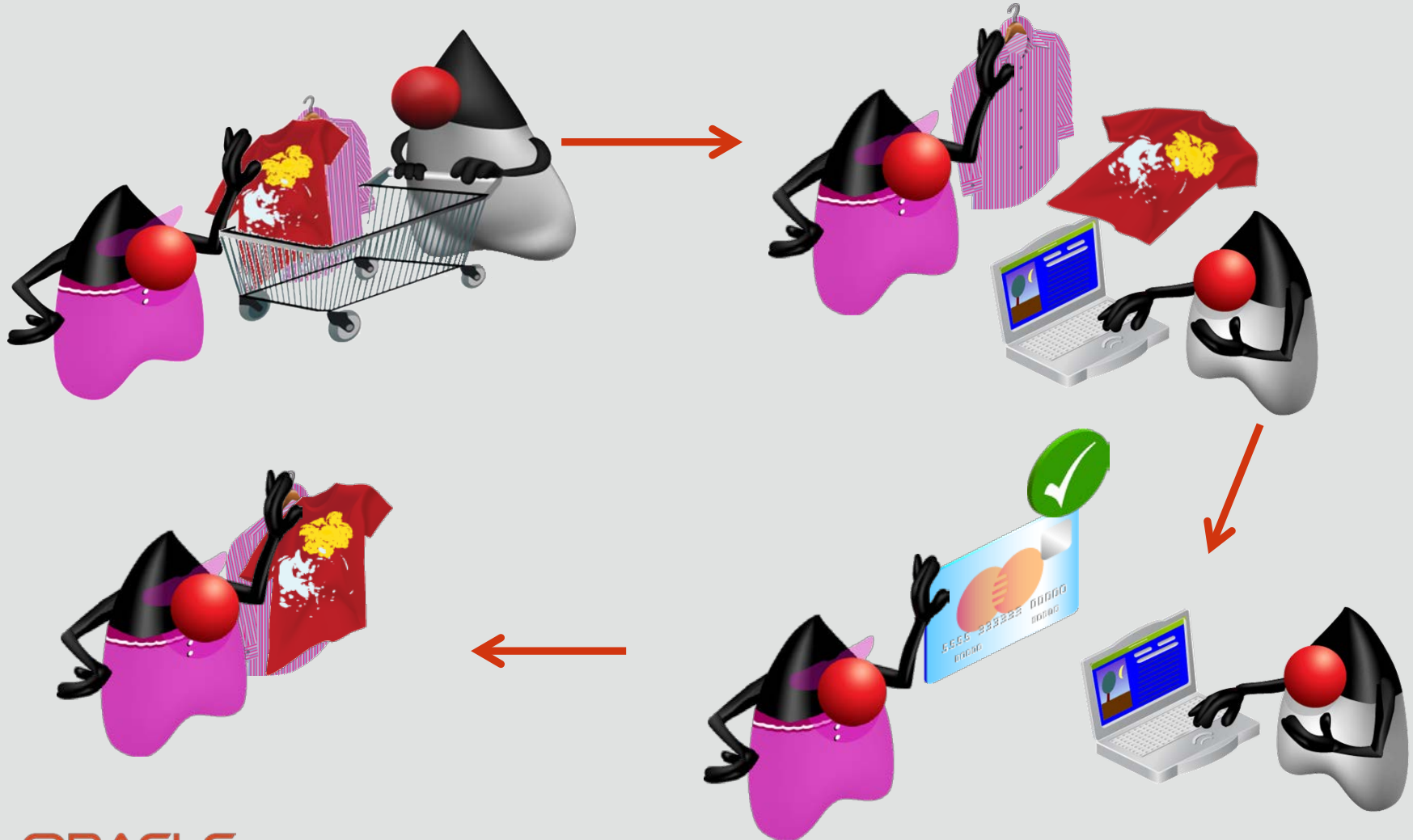
All cat instances share the ability to meow, play, and eat



Object-Oriented Strategy

- How do you write programs that achieve this level of flexibility?
- When you have an idea or requirement for a program
 - ...
 - Consider what type of objects may exist in this program
 - Consider the properties and behaviors of these object types
 - Consider how objects interact

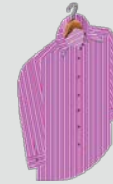
Duke's Choice Online Shopping





Characteristics of Objects

- Objects are physical or conceptual
- Objects have properties:
 - Size
 - Price
 - Color
- Objects have behaviors:
 - Shop
 - Put item in cart
 - Pay



**Physical:
Shirt**



**Conceptual:
Online
Account**



Color property value is red



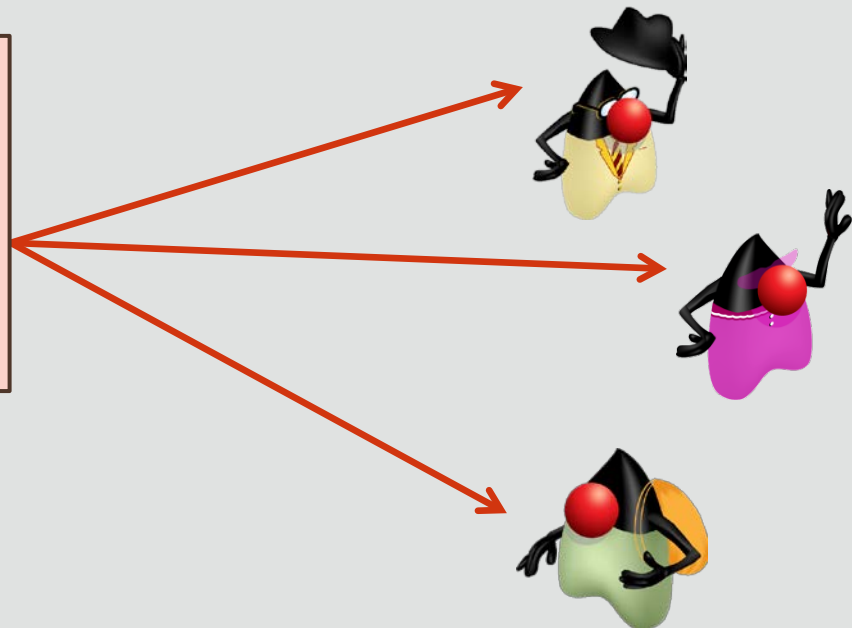
Mrs. Duke

Classes and Instances

- Remember, a class ...
 - Is a blueprint or recipe for an object
 - Describes an object's properties and behaviors
 - Is used to create Object instances



Object instances



Exercise 2, Part 1

- Given the following scenario, what objects could you potentially model to complete your program?
 - Design a program for a coin-sorting machine
 - This machine should measure, count, and sort coins based on their size or value
 - It should also print a receipt
- List at least 3 objects you would need to model for this example:
 -
 -
 -



Exercise 2, Part 2

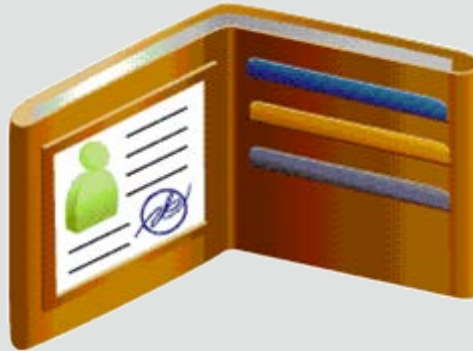
- Chose an object from Part 1
 - What properties and behaviors of this object could you include in your program?
- Properties:
 - size
 - value
 - mass
- Behaviors:
 - getSize()
 - getValue()
 - getMass()



Customer Properties and Behaviors

- Properties:

- Name
- Address
- Age
- Order number
- Customer number



- Behaviors:

- Shop
- Set address
- Add item to cart
- Ask for a discount
- Display customer details

Translating into Java Syntax

```
1 public class Customer {  
2  
3  
4     Properties  
5  
6  
7  
8     Behaviors  
9  
10  
11 }
```


Java Terminology

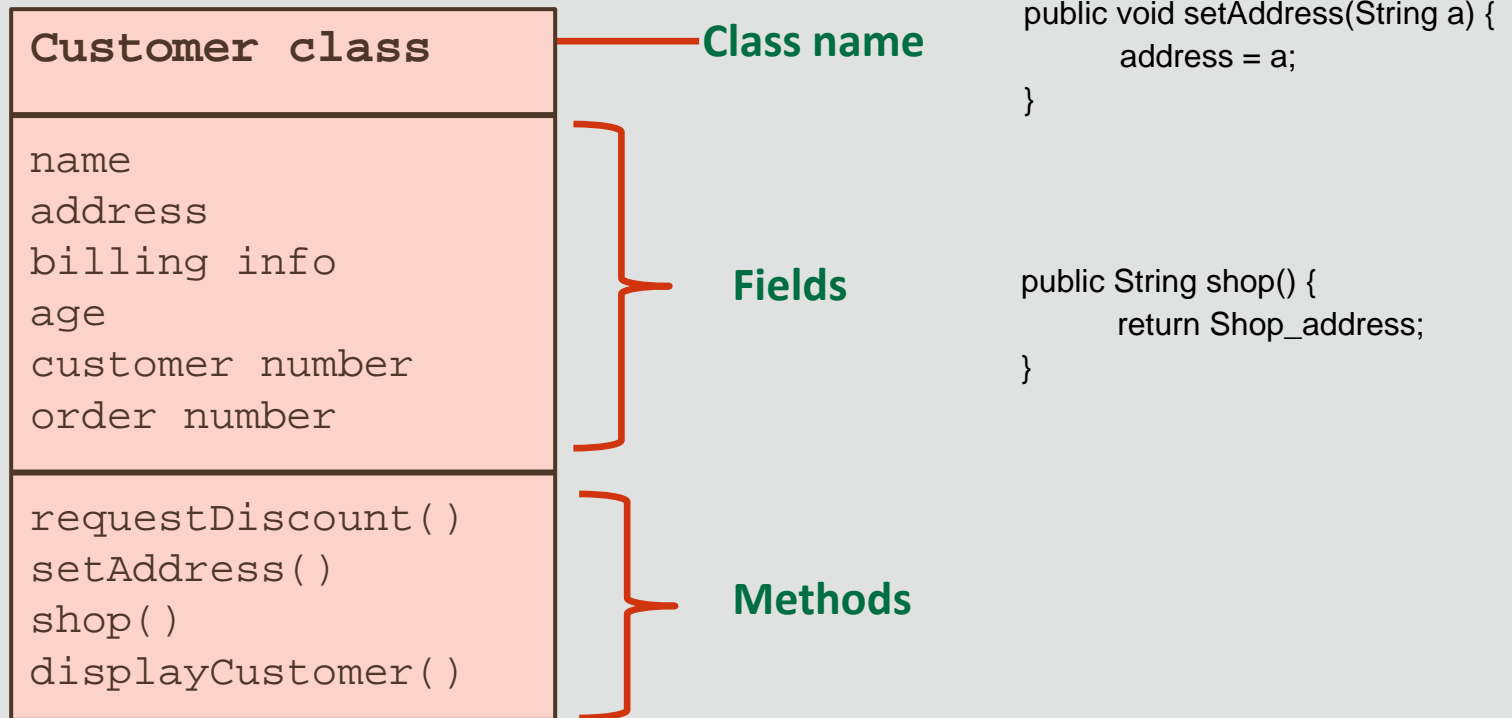
Class declaration

```
1 public class Customer {  
2     public String name = "Junior Duke";  
3     public int    custID = 1205;  
4     public String address;  
5     public int    orderNum;  
6     public int    age;  
7  
8     public void displayCustomer(){  
9         System.out.println("Customer: " + name);  
10    } //end method displayCustomer  
11 } //end class Customer
```

Fields
(Properties)
(Attributes)

Methods
(Behaviors)

Modeling Properties and Behaviors



Data Fields

- Fields or Data Fields are the official Java terminology
- They're also called:
 - Properties
 - Attributes
 - Data Members
- Java has particular ways of representing data
 - Section 3 will take a closer look at data
 - We'll use the main method for this investigation
 - For now, it's alright to include a lot of code in the main method
 - BUT a large main method is strongly discouraged
 - Section 4 explores how to avoid this scenario

Summary

- In this lesson, you should have learned how to:
 - Differentiate between procedural and object-oriented programming
 - Understand a class as a blueprint for an object
 - Understand a class is used to create instances of an object
 - Model objects as a combination of ...
 - Properties (data fields)
 - Behaviors (methods)





ORACLE

Academy

