

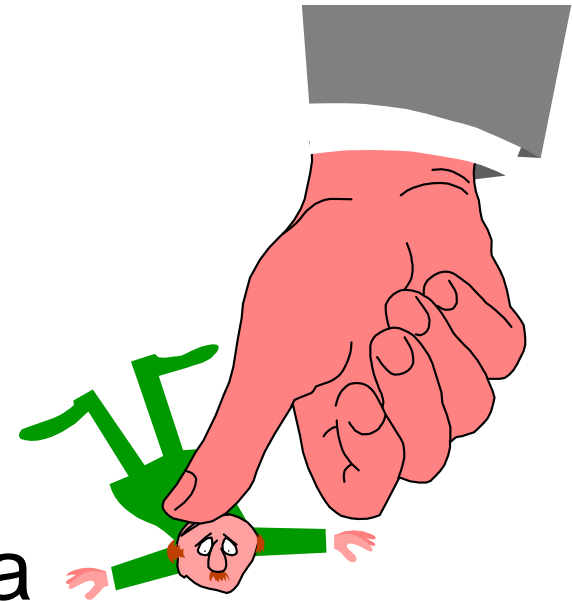
The World of Objects

Slides borrowed from **Maayan Geffet**

mary@cs.huji.ac.il

The World of Objects

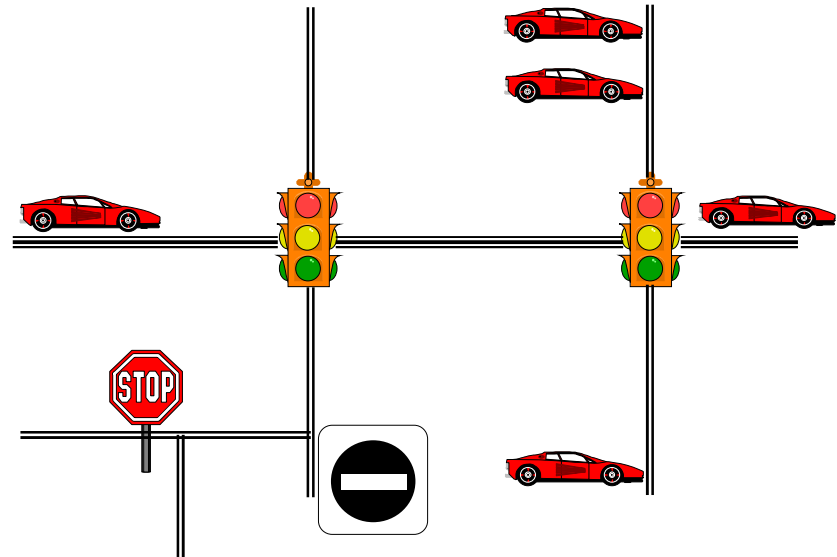
- The World consists of Objects
- Objects are Nouns
- A System of Objects is usually what we want to *simulate* or *implement* with a program on a computer



**Anything we
can put a
thumb on**

Example 1

- Traffic System
- Simulate traffic flow, traffic jams
- Objects include:
 - Cars
 - Trucks
 - Pedestrians
 - Traffic Lights
 - The Road itself!



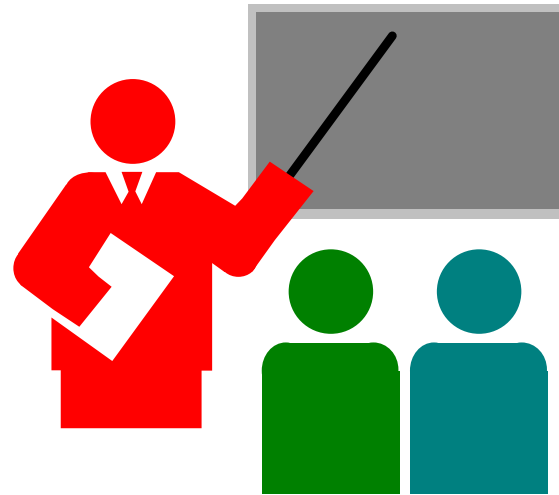
Example 2

- Checkout Counter System
- Simulate throughput at grocery store
- Objects include:
 - Customers
 - Cashiers
 - Groceries



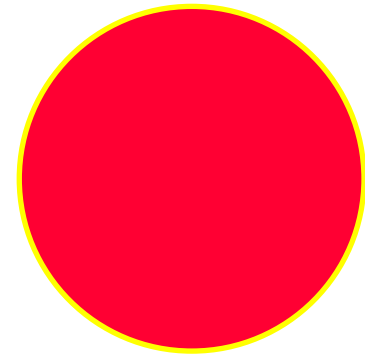
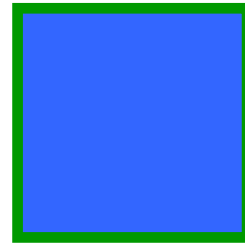
Example 3

- Class Scheduling System
- Assign students to classes
- Objects include:
 - Students
 - Classes
 - Time slots
 - Rooms



Example 4

- Graphical Drawing System
- Allow user to draw shapes and manipulate them on the screen
- Objects include:
 - Circles
 - Rectangles
 - Lines
 - Polygons



Objects have a State — Attributes



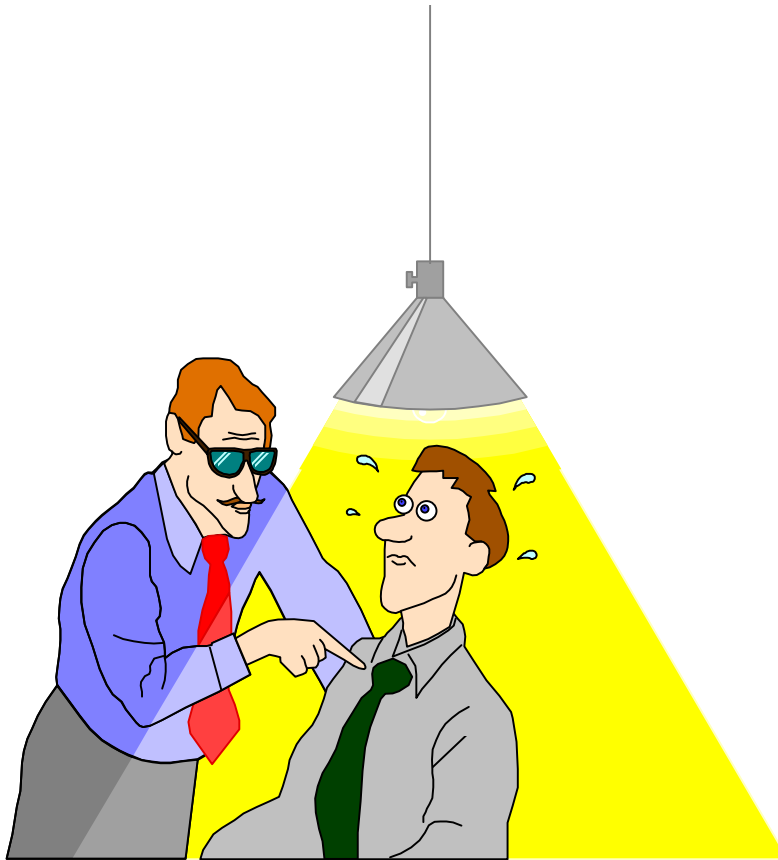
***An attribute is
any
characteristic
of an object***

Objects Can *Do Things* — *Methods*



**An object has
operations it
can perform —
built right into it**

Objects Can be Sent *Messages*



One object can ask another object for a service, by sending it a *message*

One object asks another to use a particular method

Basic Objects

- **Objects**
 - Nouns, things in the world
- **Attributes**
 - Properties each of those things have
- **Methods**
 - Actions that each of those things can do
- **Message**
 - Communication from one object to another, asking for a method to be used; the way methods are “triggered”

Example – Bank Accounts

Attributes:

owner = John Doe
balance = \$142.30
kind = Checking

Methods:

deposit()
withdraw()

Attributes:

owner = Frank Oz
balance = \$2301.47
kind = Savings

Methods:

deposit()
withdraw()

Attributes:

owner = Bill Budd
balance = \$29.88
kind = Checking

Methods:

deposit()
withdraw()

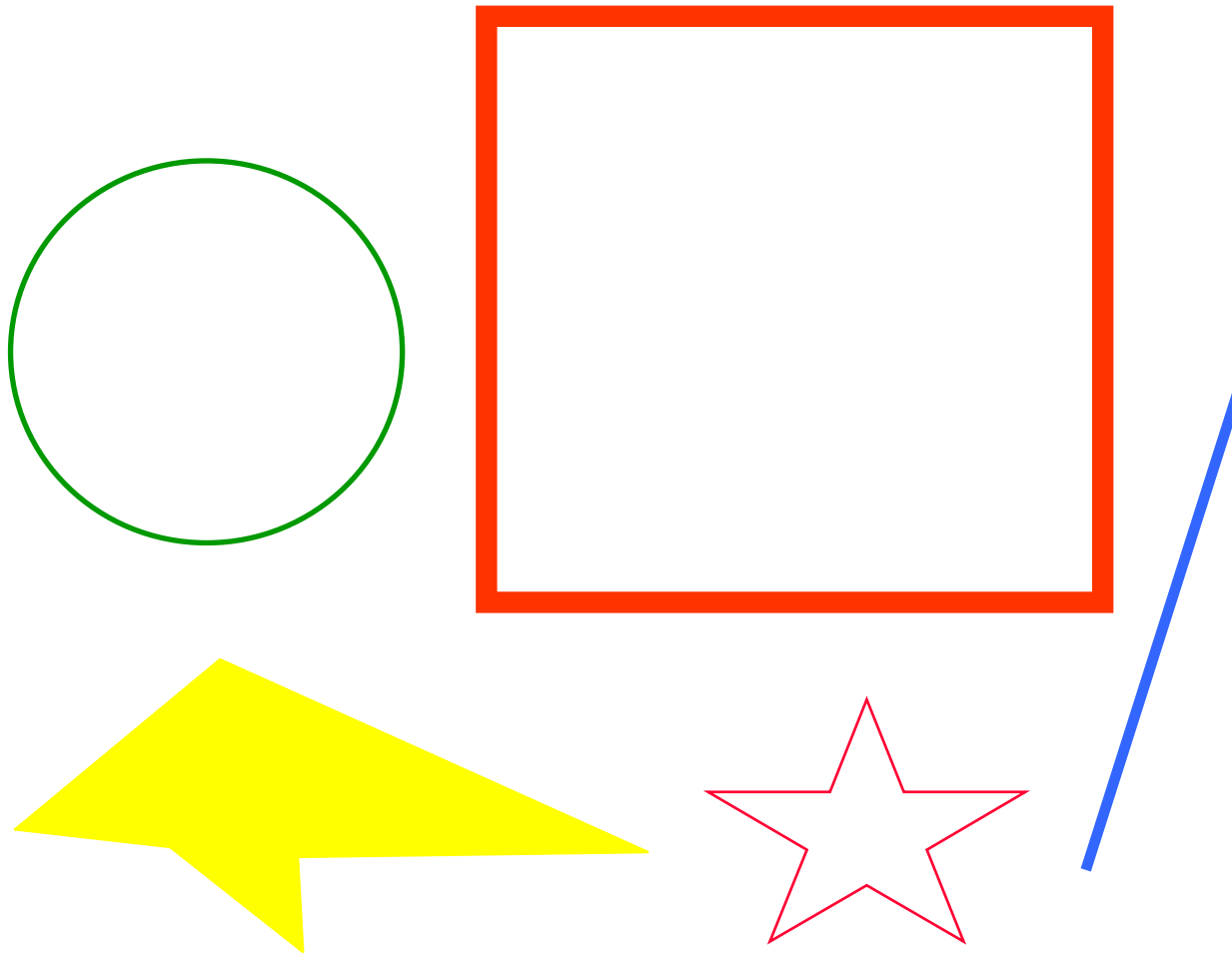
- Bank accounts have a state — attributes, like owner, balance, kind of account
- Bank accounts (in OOP) can *do* things — methods, like deposit() and withdraw()
- Each bank account is represented by an object
- Send an object a message to get it to add or subtract money

Let's Consider Shapes

- Shapes have a state — attributes
- Shapes can *do* things — methods
- Attributes of a shape:
 - Filled, line width, line color, fill color, location
- Methods of a shape:
 - Fill, Empty, Move, Grow

Fun with Shapes

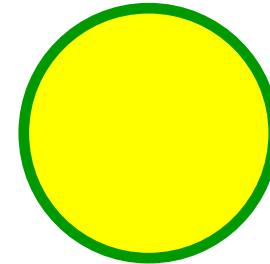
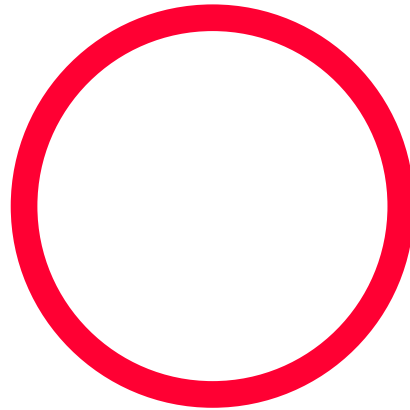
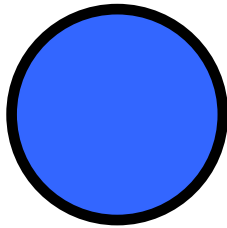
Each Shape is an Object



- **Properties of a shape:**
 - filled
 - line width
 - line color
 - fill color
 - location
- **Methods of a shape:**
 - Fill
 - Empty
 - Move
 - Grow

There is a Structure Here

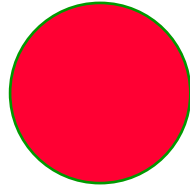
- There are certain shapes of a related kind
- This *prototype* is called a Class



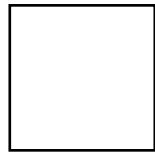
- **Each circle is different, but they are all instances of the class Circle**

Each Object is an *Instance* of a *Class*

- An Instance of the Class “Circle”



- Two Instances of the Class “Square”

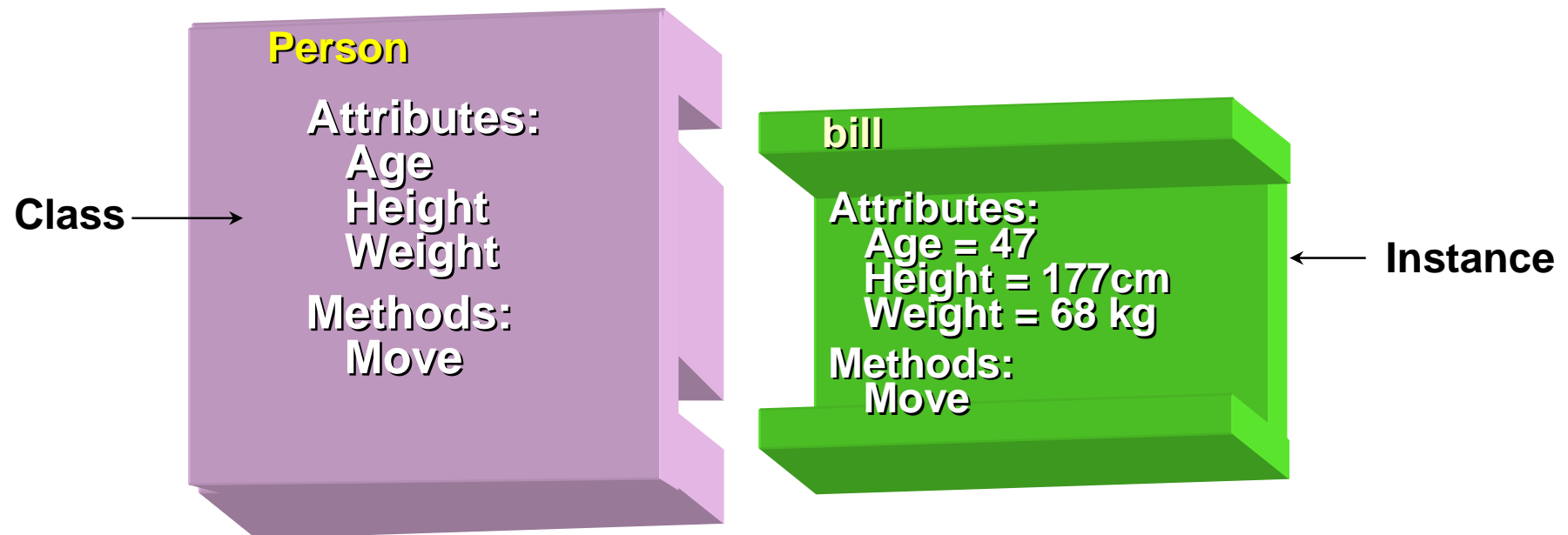


- An Instance of the Class “Line”

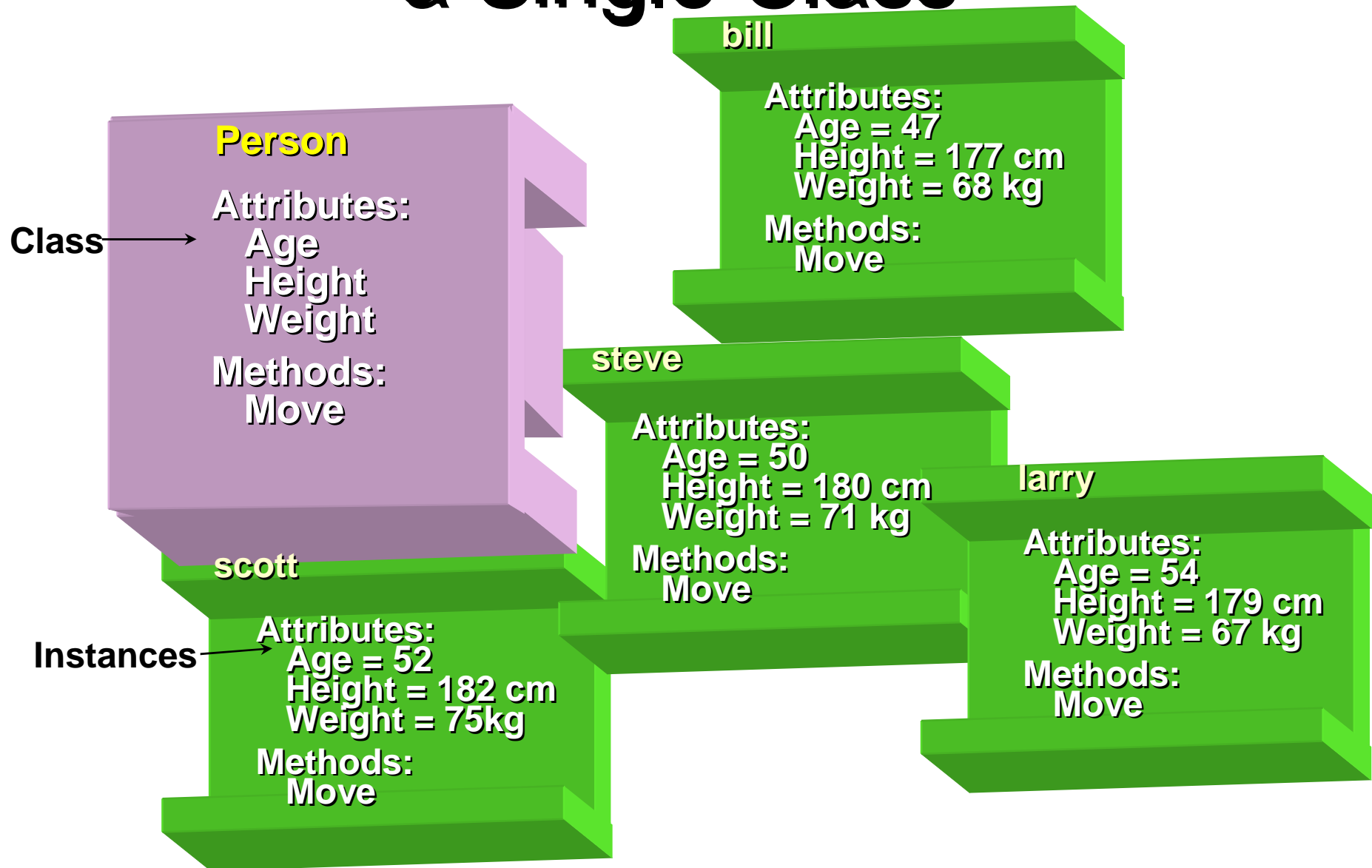


Classes

- A *Class* is an abstract description of objects having the same attributes and methods
- A specific Object is an instance of a Class
- A Class is the cookie cutter —
An Object is the cookie



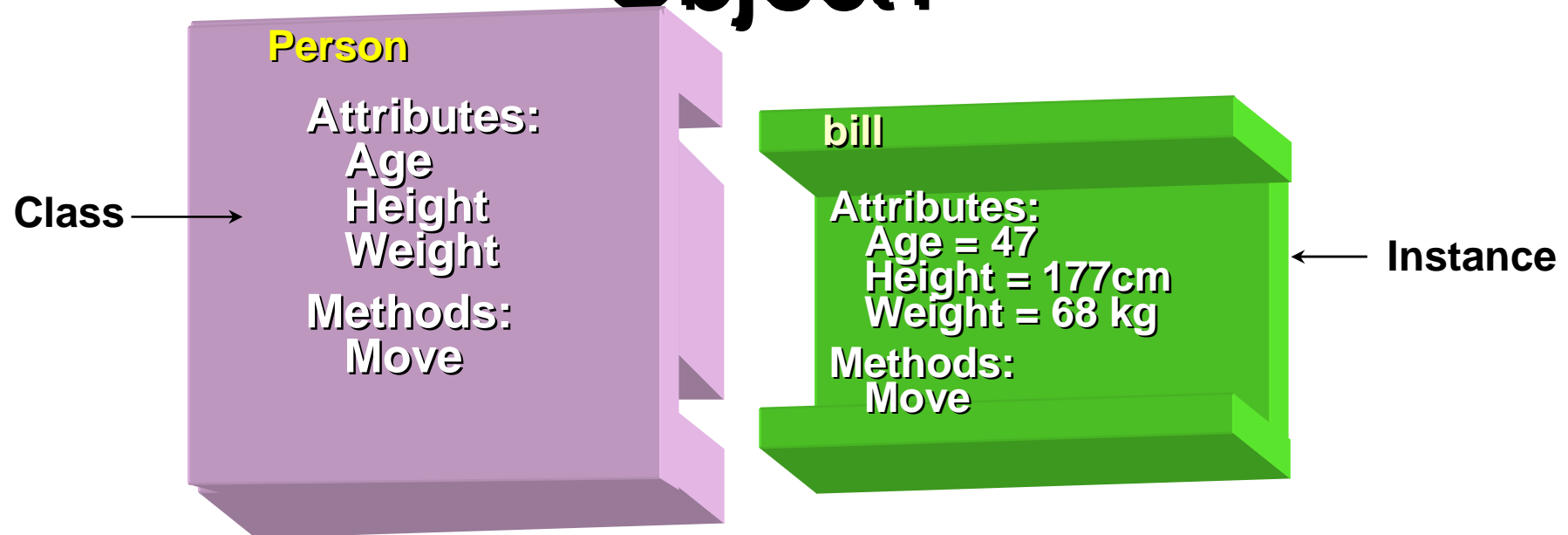
Many Different Objects from a Single Class



How Do We Create an Object?

- We use a *constructor*
- This takes a Class and creates an Instance of the class, an object, perhaps with certain properties
- “Construct an Instance of the Class Person, give it the name “bill”, and make its Age be 47, its Height be 177 cm, and its Weight be 68 kg.”

How Do We Create an Object?



- Presto! We now have an object “bill”, with certain attributes, and with the method Move
- The object “bill” can now be sent the message “Move”

Object Vocabulary

- *Classes* — Prototypes for objects
- *Objects* — Nouns, things in the world
- *Constructor* — Given a Class, the way to create an Object (that is, an *Instance* of the Class) and initialize it
- *Attributes* — Properties an object has
- *Methods* — Actions that an object can do
- *Messages* — Communication from one object to another, asking for a method to be used