Inheritance -> the topic of the week: basics, Object class
Class mechanism is brittle; Inheritance makes mechanism more flexible. Consider: Infant and WeightedInfant:

```java
public class Infant {
    private String name;
    private int age;  // in months
    public Infant(String who, int months) {
        name = who;
        age = months;
    }
    public String getName() { return name; }
    public int getAge() { return age; }
    public void anotherMonth() { age = age + 1; }
    public String toString() { return (name + " "+ age); }
}

public class WeightedInfant extends Infant {
    private double weight;
    public WeightedInfant(String who, int months, double wt) {
        super(who, months);
        weight = wt;
    }
    public double getWeight() { return weight; }
    public String toString() {
        return (super.toString() + " " + weight);
    }

    1. Make a KidDriver class: a) make a WeightedInfant object – jill/4 months/13.4 lbs
       b) print her age; 3) print her weight; 4) make her one month older; 5) Print out all of her information

    Modeling:
    2) make a Dog class (name, age); extend to a ShowDog class (showdog: yes or no?)
    3) make a Tool class (name, weight); extend to a PowerTool class (amps - double)
    4) make a Road class (start, finish, distance), include a toString method; extend to a MinorRoad class (is it paved, or not) also: add a toString method

    5) The Random class – where is it? Write a one class application that prints random ints between 10 and 19, inclusive, in a column until their sum tops 100.
Here's the shuffle method from ch 9:
public void shuffle(int[] nums)
{  
    int swapPos, temp;
    for (int i = nums.length - 1; i > 0; i--)
    {  
        swapPos = nextInt(i + 1); // pick pos:0 -> i(i possible)    temp = nums[swapPos];//
        swap vals at i, swapPos
        nums[swapPos] = nums[i];
        nums[i] = temp;
    }
}

6) How does the swap work? How would you re-write it to shuffle an array of chars?

7) Extend Random to form a Coin class. Create a main that uses Coin to flip a coin 100 times, report number of heads that come up. Hint: write the main first!!

Notes on next programming assignment 6 - read in lines of text, then print lines back, but in pig latin (words starting with vowel add “ay” at end; otherwise, consonant start – put first letter + “ay” at end). Since you need to read in all lines first, then print out complete text, in a block, in pig latin, you need to save the input lines in an array. (egg -> eggay,  boat -> oatbay). The driver:

import java.util.*;
public class PigDriver{
    public static void main(String[] args){
        Scanner scan = new Scanner(System.in);
        String t = " ";
        Piglatin p = new Piglatin();
        while(t.length() > 0){
            t = scan.nextLine();
            t = t.toLowerCase();
            p.pigConvert(t);
        }
        p.pigReport();
    }
}

Say in words what jobs pigConvert must perform. What about pigReport? What are the attributes of a Piglatin object? What library classes are needed?