Note: the OilTank class is given at the end of the exam

1. Suppose you are writing a driver class called TankDriver, which will make use of the OilTank class.
   a) In this driver class create an OilTank object called myTank, with owner Hank, with 300 gallon capacity, with 200 gallons of oil in the tank, and with an oil price of $2.59 per gallon.

   OilTank myTank = new OilTank("Hank",300,2.59,200);

   b) Write a single print statement in the driver which prints the value of the oil in myTank. (Value is the number of gallons times the price per gallon). Don't worry about money formatting.

   System.out.println(myTank.getQuantity()*myTank.getPrice());

   c) Add a new method to the OilTank class called setPrice, which is passed a new (double) price per gallon, and which sets the calling object's price attribute to this new value.

   public void setPrice(double amt){price = amt;}

   d) Add a new method to the OilTank class called halfFull, which returns true if the calling OilTank object is strictly more than half-full; otherwise the method should return false.

   public boolean halfFull(){if ((double)quantity/capacity > .5) return true; else return false;}

   e) Add a new method to the OilTank class called fillUpCost, which prints to the console the amount of money needed to fill up the tank. (Note: don't worry about proper money formatting).

   public void fillUpCost(){System.out.println( (capacity-quantity)*price);}
for(int j = 70; j >= 40; j--) System.out.println(j);

b) Using a loop, write statements that will print to the console the sum of all even numbers from 10 to 50.
    int sum = 0;
    for(int j = 10; j <= 50; j = j + 2){sum = sum + j;}
    System.out.println(sum);

c) Write a loop that will print a column of 400 random numbers between 0 and 1 to the console.
    for(int j = 0; j < 400; j++) System.out.println(Math.random());

d) Explain what this loop does:
    for(int j = 1; j != 100; j = j + 10){System.out.println(j);}
    An infinite loop: 1, 11, 21, ... it steps over 100, and keeps going

e) Write a loop that prints this pattern on a single line.
    dog cat pig dog cat pig dog cat pig ... dog cat pig. There should be 30 of each kind of animal.
    for(int j = 0; j < 30; j++)
        System.out.print("dog " + "cat " + "pig ");

3. Write a complete program that does the following: it should read in a single integer - call it n -- from the keyboard. If the number it reads is negative, the program should do nothing. Otherwise it should print, in a column, all of the integers from 1 to n that are NOT divisible by 5.

import java.util.*;

class Tester {
    public static void main(String args[]){
        Scanner s = new Scanner(System.in);
        int n = s.nextInt();
        if(n >= 0){
            for(int j = 1; j <= n; j++)
                if(j%5 != 0) System.out.println(j);
        }
    }
}

4. Short answers:
   a) Explain the difference between these two expressions, both of which use equal signs: 1) int a = 17; and
2) (a == 17)
first one an assignment stmt – copies 17 to variable a; second one is an
equality test: returns true or false, depending on value of a

b) Suppose the last line in the body of a method is:
   return (b <= 12);
   What can you conclude about the return type of the method?

   The expression (b<=12) has a boolean value, so method must have boolean return
type.

c) Explain what happens when you execute these two statements:
   String p = "kitty"; p = 12 + p;

   If one of arguments to + is a String, String concatenation is done: so p now
   holds the String 12kitty

d) The constructor of the OilTank class, below, includes the
   following statement: this.who = who; Briefly explain how this
   works.

   The Java “pronoun” “this” refers to a to-be-created object. So whenever the
   constructor is called, “this” refers to the object that is created by the call.

e) The OilTank class, below, includes six methods. Which ones can be
   used to change the state of an OilTank object?

   setQuantity and fillTank

public class OilTank{

    private int capacity; // capacity of tank
    private String who; // tank owner
    private double price; // price per gallon of oil
    private int quantity; // gallons of oil in tank

    public OilTank(String who, int cap, double price, int quant){
        this.who = who;
        capacity = cap;
        this.price = price;
        quantity = quant;
    }

    public int getCapacity(){return capacity;}
    public String getWho(){return who;}
    public double getPrice(){return price;}
}
public int getQuantity(){return quantity;}

public void setQuantity(int amt){ quantity = amt;}

public void fillTank(){quantity = capacity; }
}