1. Write a Pet class. A Pet object has the following variables: String name, String animalType, and int age. Write a constructor that accepts the parameters in that order. Write a toString which returns a String formatted like this: "Gilly, dog, age: 8".

```
public Pet
{
    private String name;
    private String animalType;
    private int age;
    public Pet (String n, String at, int a)
    {
        name = n; animalType = at; age = a;
    }
    public String toString()
    {
        return name + ", " + animalType + ", age: " + age;
    }
}
```

2. What is output by this code?	3. What is output by this code?

`String str = "I love the rain";`
(annotated above: 0 1 2 3 4 5 over "I love")
`System.out.println(str.indexOf("e"));`
 	
5
	`String str = "I love the rain";`
	`System.out.println(str.substring(2,5));`
	
	lov ✓

4. Write code to store a random integer between 4 and 13 inclusive in an int variable named **num**:   13-4=9 +1 ≤10 ↙
 	
`int num = (int)(Math.random()*10)+4;`
	5. Write a for loop that prints the numbers 7 through 107 going up by 10s (i.e. 7 17 27...)
	
	`for(int i = 7; i < 108; i+=10)`
	  `System.out.println(i);`

6. Write a while loop which prints the numbers 9 down to zero:
 	
`int n = 9;`
`while (n >= 0)`
`{`
  `System.out.println(n);`
  `n--;`
`}`
	7. Assume an int variable temp contains valid data ~~and a boolean variable hot that is set to false~~. Write an if / else if / else statement which checks if temp is greater than 90 (print "yes"), between 70 and 90 inclusive (print "OK") and otherwise prints "no".
	
	`if (temp > 90)`
	  `System.out.println("yes");`
	`else if (temp >= 70 && temp <= 90)`
	  `System.out.println("OK");`
	`else`
	  `System.out.println("no");`

8. Two String variables **answer1** and **answer2** contain valid data. Write an if statement that checks if they are equal. Print "yes" if they are, do nothing if they are not.
 	
`if(answer1.equals(answer2))`
  `System.out.println("yes");`	

9. Tourist class: A tour company keeps track of tourists using Java and a Tourist class. The Tourist class has at least these variables: String firstName, String originCountry, boolean speaksEnglish, and int seatNumber, which contains a seat number (for the bus) for each tourist, starting at 10, going up by 1.

The constructor accepts the parameters in this order: firstName, originCountry, and speaksEnglish and then creates a Tourist with the given information and a unique seat number.

The boolean checkLanguage() method returns the value of speaksEnglish.
The void newSeat() method changes the tourist's seat to the next available bus seat.
The String toString method returns a String representation of a Tourist that looks like this:

```
Seat #10: Frankie, from Bolivia  (English OK)
Seat #11: Franny, from Spain (does not speak English)
```

Write the entire Tourist class here.

```java
public Tourist
{
    private String firstName;
    private String originCountry
    private boolean speaksEnglish;
    private int seatNumber;
    private static int nextSeat = 10;

    public Tourist (String n, String c, boolean e)
    {
        firstName = n;
        originCounty = c;
        speaksEnglish = e;
        seatNumber = nextSeat;
        nextSeat++;
    }

A   public boolean checkLanguage()
    {
        return speaksEnglish;
    }

M   public void newSeat()
    {
        seatNumber = nextSeat;
        nextSeat++;
    }

    public String toString()
    {
        String eng = "(does not speak English)";
        if (speaksEnglish)
            eng = "(English OK)";
        return "Seat #" + seatNumber +
            ": " + firstName + ", from " +
            originCountry + " " + eng;
    }
}
```

When done, put a big M on the left next to the mutator method and a big A next to the accessor method.

Write code here to create a Tourist, then to print the Tourist, and then to change the Tourist's seat.

```java
Tourist X = new Tourist ("Malcom", "Argentina", false);
System.out.println(X);
X.newSeat();
```