Boot Camp Day-4

No	Quiz/ Question	Input Data	Output	Hint
1	Write a java program to create an abstract class named Shape that contain an empty method named numberOfSides() and area(). Provide three classes named Trapezoid, Triangle and Hexagon such that each one of the classes extends the class Shape. Each one of the class contains only the method numberOfSides() that shows the number of sides in the given geometrical figures.	Create a menu for the shapes. Based on the option selected, create object of the shape and display the output of the two abstract	Print number of sides and area for the selected shape.	Use abstract concept
2	A class Connection maintains attributes database url, user name and password. Class needs to maintain the number of count of Connection class. Every time a connection object is created the count must be incremented and every time it is set to null count must be decremented and object must be garbage collected explicitly by the code. At any point, there must be only 10 connection object in the memory. Write a java class to achieve this.	Read connection attributes.	User information: Number of connections:	Hint: use finalize() and System.gc()
3	Create a class that represents a time in the form of hours, minutes and seconds. Accept the time parameters from the user in the form of a. hours, minutes and seconds b. hh:mm:ss -Only valid time must be allowed to be created. -Display the time in the form of hh:mm:ss.	Input time based on the given form:	6:20:45	
4	Create a class Employee that stores data about employee consisting of name, employee code , gender, location (East, West, North, South). Whenever new employee object is created, employee code is automatically generated. The employee code is a combination of the first letter of the location and sequence number. The sequence number start from 0001 and is incremented till 9999, after which the application must generate an error. Write a test program to demonstrate all the capabilites.	Read employee information:	Show employee information. For testing the program keep the sequence limit as 0001 to 0005	
5	Write a java program to create an immutable class and create a test program to use the services of the class.			Check for immutable classes definition

Boot Camp Day-4

No	Quiz/ Question	Input Data	Output	Hint
6	Different banks use different methods to calculate inrerest. Create a parent class named Bank with one abstract method called calculate_Interest() and derive SBI, HDFC and BOB classes with necessary members added to these classes. Using dynamic method dispatch concept create minimum one object for each derive class and print the customer information. use the following to calculate the new balance: new_balance=balance+interest.	Input account_no, customer_name, trans_type and balance amount:	Account no: 12122 Customer name: Raj Bank name: SBI Transaction type: Deposit Transaction amount: Rs. 7000.00 Balance amount: Rs. 25000.00 Interest amount: Rs. 25000+interest(as per the formula used in the bank) New Balance: Rs. 32000.00	Use runtime polymophism.
7	Write a java program to demonstrate this and super keywords. In this problem we have given you three classes in the editor: Student class Trainer class HCL class In the main method, we populated an ArrayList with several instances of these classes. count method calculates how many instances of each type is present in the ArrayList. The code prints three integers, the number of instance of Student class, the number of instance of Trainer class, the number of instance of Hcl class. But some lines of the code are missing, and you have to fix it by modifying only lines! Don't add, delete or modify any extra line.	Sample Input 5 Student Student Trainer Student HCL	Sample Output 3 1 1	Use keywords *super* and *this* . **Do not use array of any class to implement this.**

Code:

```
import java.util.*;

class Student{}
class Trainer{   }
class Hcl{}
public class InstanceOFTutorial{
   static String count(ArrayList mylist){
      int a = 0,b = 0,c = 0;
      for(int i = 0; i < mylist.size(); i++){
         Object element=mylist.get(i);
         if(~~Complete this line~~)   a++;
         if(~~Complete this line~~)   b++;
         if(~~Complete this line~~)   c++;
      }
```

Boot Camp Day-4

No	Quiz/ Question	Input Data	Output	Hint

	String ret = Integer.toString(a)+" "+
	Integer.toString(b)+" "+ Integer.toString(c);
	return ret;
	}
	
	public static void main(String []args){
	ArrayList mylist = new ArrayList();
	Scanner sc = new Scanner(System.in);
	int t = sc.nextInt();
	for(int i=0; i<t; i++){
	String s=sc.next();
	if(s.equals("Student"))
	mylist.add(new Student());
	if(s.equals("Trainer"))
	mylist.add(new Trainer());
	if(s.equals("Hcl"))
	mylist.add(new Hcl());
	}
	System.out.println(count(mylist));
	}
	}			

	Following is an example of abstract class:
	
	abstract class Book{
	String title;
	abstract void setTitle(String s);
	String getTitle(){
	return title;
	}
	}
	If you try to create an instance of this class like the following line you will get an error:
	
	Book new_novel=new Book();
	
	You have to create another class that extends the abstract class. Then you can create an instance of the new class.
	Notice that *setTitle* method is abstract too and has no body. That means you must implement the body of that method in the child class.
	
	In the editor, we have provided the abstract Book class and a Main class. In the Main class, we created an instance of a class called MyBook. Your task is to write just the MyBook class.
	
	Your class mustn't be public.
		Sample Input
		
		A tale of two cities
			Sample Output
			
			The title is: A tale of two cities	

Boot Camp Day-4

No	Quiz/ Question	Input Data	Output	Hint

9	import java.util.*;
	abstract class Book{
	String title;
	abstract void setTitle(String s);
	String getTitle(){
	return title;
	}
	}
	
	//Write MyBook class here
	
	public class Main{
	public static void main(String []args){
	//Book new_novel=new Book(); This line prHMain.java:25: error: Book is abstract; cannot be instantiated
	Scanner sc=new Scanner(System.in);
	String title=sc.nextLine();
	MyBook new_novel=new MyBook();
	new_novel.setTitle(title);
	System.out.println("The title is: "+new_novel.getTitle());
	sc.close();
	
	}
	}
				Abstract class

	You are given a partially completed code in the editor. Modify the code so that the code prints the following text:
	
	Hello I am a motorcycle, I am a cycle with an engine.
	My ancestor is a cycle who is a vehicle with pedals.
	
	import java.util.*;
	import java.io.*;
	class BiCycle{
	String define_me(){
	return "a vehicle with pedals.";
	}
	}
	
	class MotorCycle extends BiCycle{			

Boot Camp Day-4

No	Quiz/ Question	Input Data	Output	Hint

10	String define_me(){
	return "a cycle with an engine.";
	}
	
	MotorCycle(){
	System.out.println("Hello I am a "
	" motorcycle, I am "+ define_me());
	String temp=define_me(); //Fix this line
	System.out.println("My ancestor is a cycle "+
	"who is "+ temp);
	}
	}
	class Solution{
	public static void main(String []args){
	MotorCycle M=new MotorCycle();
	}
	}			