Lab Assignment 07

Inspiring Excellence

Course Code:	**CSE111**
Course Title:	**Programming Language II**
Topic:	**Array Manipulation and Inheritance Basics**
Number of Tasks:	**9**

[Submit all the Coding Tasks (Task 1 to 8) in the Google Form shared on buX before the next lab. Submit the Tracing Tasks (Task 9) handwritten to your Lab Instructors at the beginning of the lab]

[You are not allowed to change the driver/given codes of any of the tasks]

Task 1

Design the **BrowserHistory** class so that the given output is produced.

Driver Code	Output

public class BrowserHistoryTester {
  public static void main(String[] args){
  BrowserHistory chrome = new BrowserHistory(3);
  System.out.println(chrome.count + " web pages visited.");
  System.out.println("1=========");
  chrome.showHistory();
  System.out.println("2=========");
  chrome.currentWeb();
  System.out.println("3=========");
  chrome.visitWeb("www.google.com");
  chrome.visitWeb("www.facebook.com");
  chrome.visitWeb("www.youtube.com");
  chrome.visitWeb("www.reddit.com");
  System.out.println("4=========");
  System.out.println(chrome.count + " web pages visited.");
  System.out.println("5=========");
  chrome.showHistory();
  System.out.println("6=========");
  chrome.currentWeb();
  System.out.println("7=========");
  chrome.back();
  System.out.println("8=========");
  System.out.println(chrome.count + " web pages visited.");
  System.out.println("9=========");
  chrome.currentWeb();
  System.out.println("10=========");
  chrome.back();
  System.out.println("11=========");
  chrome.currentWeb();
  System.out.println("12=========");
  chrome.back();
  System.out.println("13=========");
  chrome.currentWeb();
  System.out.println("14=========");
  System.out.println(chrome.count + " web pages visited.");
  }
}
	0 web pages visited.
	1=========
	No web pages visited yet.
	2=========
	You have not visited any website yet!
	3=========
	Visited: www.google.com
	Visited: www.facebook.com
	Visited: www.youtube.com
	History is full. Cannot visit
	www.reddit.com web page.
	4=========
	3 web pages visited.
	5=========
	Browser History:
	www.google.com
	www.facebook.com
	www.youtube.com
	6=========
	Current web page: www.youtube.com
	7=========
	8=========
	2 web pages visited.
	9=========
	Current web page: www.facebook.com
	10=========
	11=========
	Current web page: www.google.com
	12=========
	13=========
	You have not visited any website yet!
	14=========
	0 web pages visited.

Task 2

Design the **CustomerService** class so that the given output is produced.

Driver Code	Output

public class CustomerServiceTester {
  public static void main(String[] args) {
  CustomerService service = new CustomerService(3);
  service.serveCustomer();
  System.out.println("1=========");
  service.addCustomer("Arthur");
  service.addCustomer("Bruce");
  service.addCustomer("Clark");
  service.addCustomer("Kara");
  System.out.println("2=========");
  service.serveCustomer();
  service.serveCustomer();
  System.out.println("3=========");
  service.addCustomer("Diana");
  service.addCustomer("Victor");
  service.addCustomer("Harley");
  System.out.println("4=========");
  service.serveCustomer();
  service.serveCustomer();
  service.serveCustomer();
  service.serveCustomer();
  }
}
	No customers to serve
	1=========
	Added: Arthur
	Added: Bruce
	Added: Clark
	Queue is full. Cannot add Kara
	2=========
	Serving Arthur
	Serving Bruce
	3=========
	Added: Diana
	Added: Victor
	Queue is full. Cannot add Harley
	4=========
	Serving Clark
	Serving Diana
	Serving Victor
	No customers to serve

Task 3

Complete the class **Circle** so that the desired outputs are generated properly.

Given Code	Expected Output

`public class shapeTester {`
`  public static void main(String[] args) {`
`  Shape s = new Shape();`
`  s.name = "Mobius Strip";`
`  s.color = "Blue";`
`  s.displayInfo();`
`  System.out.println("======================");`
`  Circle c = new Circle();`
`  System.out.println("======================");`
`  c.name = "Circle";`
`  c.color = "Red";`
`  c.radius = 5;`
`  c.displayInfo();`
`  System.out.println("======================");`
`  c.area();`
`  }`
`}`
 	
`public class Shape {`
`  public String name;`
`  public String color;`
 	
`  public void displayInfo() {`
`  System.out.printf("Name: %s\nColor: %s\n", name, color);`
`  }`
`}`
 	
`public class Circle extends Shape {`
`  //Your Code Here`
`}`
	`Name: Mobius Strip`
	`Color: Blue`
	`======================`
	`======================`
	`Name: Circle`
	`Color: Red`
	`======================`
	`Area of Red Circle: 78.54`

Task 4

Complete the class **Dog** so that the desired outputs are generated properly.

Given Code	Expected Output

`public class AnimalTester{`
`  public static void main(String args[]){`
`  Animal a1 = new Animal();`
`  System.out.println("1-------------");`
`  a1.details();`
`  System.out.println("2-------------");`
`  Dog d1  = new Dog();`
`  d1.name = "Pammy";`
`  System.out.println("3-------------");`
`  System.out.println("Name: " + d1.getName());`
`  d1.details();`
`  System.out.println("4-------------");`
`  d1.updateSound("Bark");`
`  System.out.println("5-------------");`
`  d1.details();`
`  }`
`}`
 	
`public class Animal{`
`  public int legs = 4;`
`  public String sound = "Not defined";`
 	
`  public void details(){`
`  System.out.println("Legs: "+legs);`
`  System.out.println("Sound: "+sound);`
`  }`
`}`
 	
`public class Dog extends Animal{`
`  //Your Code Here`
`}`
	`1-------------`
	`Legs: 4`
	`Sound: Not defined`
	`2-------------`
	`The dog says hello!`
	`3-------------`
	`Name: Pammy`
	`Legs: 4`
	`Sound: Not defined`
	`4-------------`
	`5-------------`
	`Legs: 4`
	`Sound: Bark`

Task 5

Given the following classes, write the code for the **Cricket_Tournament** and the **Tennis_Tournment** classes derived from **Tournament** so that the following output is generated.

Given Code	Expected Output

`public class Tester5 {`
`  public static void main(String[] args) {`
`  Cricket_Tournament ct1 = new Cricket_Tournament();`
`  System.out.println(ct1.detail());`
`  System.out.println("----------");`
 	
`  Cricket_Tournament ct2 = new Cricket_Tournament("IPL", 10, "t20");`
`  System.out.println(ct2.detail());`
`  System.out.println("----------");`
 	
`  Tennis_Tournament tt = new Tennis_Tournament("Roland Garros", 128);`
`  System.out.println(tt.detail());`
`  }`
`}`
 	
`public class Tournament {`
`  private String name;`
`  public Tournament() {`
`  this.name = "Default";`
`  }`
`  public Tournament(String name) {`
`  this.name = name;`
`  }`
`  public void setName(String name){`
`  this.name = name;`
`  }`
`  public String getName(){`
`  return this.name;`
`  }`
`}`
	Cricket Tournament Name: Default
	Number of Teams: 0
	Type: No type
	---------- 	
	Cricket Tournament Name: IPL
	Number of Teams: 10
	Type: t20
	---------- 	
	Tennis Tournament Name: Roland Garros
	Number of Players: 128

Task 6

Given the following classes, write the code for the **Book** and the **CD** class so that the following output is printed.

Given Code	Expected Output

public class Tester6 {
  public static void main(String[] args) {
  Book book = new Book(1, "The Alchemist", 500, "97806",
"HarperCollins");
  System.out.println(book.printDetail());
  System.out.println("----------");
 	
  CD cd = new CD(2, "Shotto", 300, "Warfaze", 50, "Hard Rock");
  System.out.println(cd.printDetail());
  }
}
 	
class Product {
  private int id;
  private String title;
  private int price;
 	
  public Product(int id, String title, int price) {
  this.id = id;
  this.title = title;
  this.price = price;
  }
 	
  public String getIdTitlePrice() {
  return "ID: " + id + " Title: " + title + " Price: " + price;
  }
}
	ID: 1 Title: The Alchemist Price: 500
	ISBN: 97806 Publisher: HarperCollins
	---------- 	
	ID: 2 Title: Shotto Price: 300
	Band: Warfaze Duration: 50 minutes
	Genre: Hard Rock

Task 7

Given the following classes, write the code for the **CSEStudent** class derived from **Student** so that the following output is generated.

Given Code

```
public class StudentTester{
  public static void main (String args[]){
    CSEStudent.details();
    System.out.println("1--------------");
    CSEStudent s1 =  new CSEStudent("Bob", 23);
    s1.info();
    System.out.println("2--------------");
    CSEStudent s2 =  new CSEStudent("Don", 33);
    s2.info();
    System.out.println("3--------------");
    s1.addLabBasedCourse("CSE220");
    s1.addLabBasedCourse("CSE221");
    System.out.println("4--------------");
    s1.info();
    System.out.println("5--------------");
    CSEStudent.details();
    System.out.println("6--------------");
    s1.addLabBasedCourse("CSE230");
    System.out.println("7--------------");
    s1.info();
    System.out.println("8--------------");
    s2.addLabBasedCourse("CSE110");
    s2.info();
  }
}

class Student{
  public String name;
  public int id;
  public String courses = "";

  public Student(String n, int i){
    name = n;
    id = i;
  }

  public void info(){
    System.out.println("Name: "+name);
    System.out.println("ID: "+id);
    System.out.println("Courses: "+courses);
  }
}
```

Expected Output

```
Total CSE Students: 0
Available Lab Based Courses:
CSE110 CSE111 CSE220 CSE221
1--------------
Name: Bob
ID: 23
Courses:
2--------------
Name: Don
ID: 33
Courses:
3--------------
4--------------
Name: Bob
ID: 23
Courses: CSE220 CSE221
5--------------
Total CSE Students: 2
Available Lab Based Courses:
CSE110 CSE111 CSE220 CSE221
6--------------
It is not a lab based course!
7--------------
Name: Bob
ID: 23
Courses: CSE220 CSE221
8--------------
Name: Don
ID: 33
Courses: CSE110
```

Task 8

Given the following classes, write the code for the **Player** and the **Manager** classes derived from SportsPerson class so that the following output is printed. To calculate the match earnings use the following formula:

1. Player: (total_goal * 1000) + (total_match * 10)
2. Manager: match_win * 1000

Given Code	Expected Output

public class PlayerTester {
  public static void main(String[] args) {
  Player playerOne = new Player("Al-Nassr", "Ronaldo", "Striker", 25, 32);
  playerOne.calculateRatio();
  playerOne.printDetails();
  System.out.println("----------------");
  Manager managerOne = new Manager("Real Madrid", "Zidane", "Manager", 25);
  managerOne.printDetails();
  }
}
 	
class SportsPerson {
  private String team;
  private String name;
  public String role;
  public double earningPerMatch;
 	
  public SportsPerson(String teamName, String name, String role){
  this.team = teamName;
  this.name = name;
  this.role = role;
  this.earningPerMatch = 0;
  }
 	
  public String getNameTeam() {
  return "Name: " + name + ", Team Name: " + team;
  }
}
	Name: Ronaldo, Team Name: Al-Nassr
	Team Role: Striker
	Total Goal: 25, Total Played: 32
	Goal Ratio: 0.78
	Match Earning: 25320K
	---------- 	
	Name: Zidane, Team Name: Real Madrid
	Team Role: Manager
	Total Win: 25
	Match Earning: 25000K

Task 9

```
1  public class A{
2    public int temp = 4;
3    public int sum = 1;
4    public int y = 2;
5    public void methodA(int m, int n){
6      int x = 0;
7      y = y + m + (temp++);
8      x = x + 2 +  n;
9      sum = sum + x + y;
10     System.out.println(x + " " + y+ " " + sum);
11   }
12 }
13 public class B extends A {
14   public int x = 6;
15   public void methodB(int m, int n){
16    int  y =0;
17    y = y + this.y;
18    x = this.y + 2 + temp;
19    methodA(x, y);
20    sum = x + y + super.sum;
21    System.out.println(x + " " + y+ " " + sum);
22   }
23 }
```

```
A a1 = new A();
a1.methodA(1, 1);
B b1 = new B();
b1.methodB(1, 2);
```

x	y	sum

Ungraded Tasks (Optional)

(You don't have to submit the ungraded tasks)

Task 1

Complete the class **Student** so that the desired outputs are generated properly.

Given Code	Expected Output

```
public class Tester5 {
  public static void main(String[] args) {
    Student s1= new Student(1, "Matthew", 1, "Mr. XYZ");
    s1.printStudentInfo();
    System.out.println("======================");
    Student s2= new Student(2, "Wade", 1, "Mr. XYZ");
    s2.printStudentInfo();
    System.out.println("======================");
    Student s3= new Student(1, "Logan", 2, "Mr. ABC");
    s3.printStudentInfo();
  }
}


public class Section {
  public int sec_id;
  public String teacher_name;
  public int student_id;
  public String student_name;
  public Section(int id, String name){
    this.sec_id = id;
    this.teacher_name = name;
  }

  public void printInfo(){
    System.out.println("Section id: "+ this.sec_id);
    System.out.println("Teacher name: "+ this.teacher_name);
  }
}

class Student extends Section {
  //Your Code Here

}
```

```
Student id: 1
Student name: Matthew
Section id: 1
Teacher name: Mr. XYZ
======================
Student id: 2
Student name: Wade
Section id: 1
Teacher name: Mr. XYZ
======================
Student id: 1
Student name: Logan
Section id: 2
Teacher name: Mr. ABC
```

Task 2

Complete the class Section so that the desired outputs are generated properly.

Given Code

```
public class Tester6 {
  public static void main(String[] args) {
    Section sec1= new Section("Mr. XYZ");
    Section sec2= new Section("Mr. ABC");
    System.out.println("Number of Sections: "+ Section.section_count);
    System.out.println("======================");
    Student s1= new Student(1, "Matthew", sec1.sec_id,
sec1.teacher_name);
    s1.printStudentInfo();
    System.out.println("======================");
    Student s2= new Student(2, "Wade", sec1.sec_id,
sec1.teacher_name);
    s2.printStudentInfo();
    System.out.println("======================");
    Student s3= new Student(1, "Logan", sec2.sec_id,
sec2.teacher_name);
    s3.printStudentInfo();
    System.out.println("======================");
    System.out.println("Number of Students: "+ Student.student_count);
  }
}


public class Section {
  //Your Code Here
}

public class Student extends Section {
  public Student (int id, String name, int sec_id, String
teacher_name){
    student_count+=1;
    this.student_id = id;
    this.student_name = name;
    this.sec_id= sec_id;
    this.teacher_name= teacher_name;
  }

  public void printStudentInfo(){
    System.out.println("Student id: "+ this.student_id);
    System.out.println("Student name: "+ this.student_name);
    System.out.println("Section id: "+ this.sec_id);
    System.out.println("Teacher name: "+ this.teacher_name);
  }
}
```

Expected Output

```
Number of Sections: 2
======================
Student id: 1
Student name: Matthew
Section id: 1
Teacher name: Mr. XYZ
======================
Student id: 2
Student name: Wade
Section id: 1
Teacher name: Mr. XYZ
======================
Student id: 1
Student name: Logan
Section id: 2
Teacher name: Mr. ABC
======================
Number of Students: 3
```