Write the code in java for the **“Vehicle”** class. The tester class and the output is given below:

| **Tester class** | **Output** |
| --- | --- |
| public class TesterVehicle{  public static void main(String [] args){  Vehicle car = new Vehicle();  System.out.println("Attributes of car object:");  System.out.println(car.type);  System.out.println(car.wheels);  System.out.println(car.color);  System.out.println("=========");  Vehicle bike = new Vehicle();  bike.type="Motor bike";  bike.wheels=2;  bike.color="Red";  System.out.println("Attributes of bike object:");  System.out.println(bike.type);  System.out.println(bike.wheels);  System.out.println(bike.color);  }  } | Attributes of car object:  Car  4  White  =========  Attributes of bike object:  Motor bike  2  Red |

| **Driver Code** | **Output** |
| --- | --- |
| public class TesterCourse{  public static void main(String[] args) {  Course c1 = new Course();  Course c2 = new Course();    System.*out*.println("========== 1 ==========");  c1.createCourse("Programming Language I", "CSE110", 3);  c1.displayCourse();    System.*out*.println("========== 2 ==========");  c2.createCourse("Data Structures", "CSE220", 3);  c2.displayCourse();    System.*out*.println("========== 3 ==========");  c1.updateCourse("Programming Language II", "CSE111", 3);  c1.displayCourse();  }  } | ========== 1 ==========  Course Name: Programming Language I  Course Code: CSE110  Course Credit: 3  ========== 2 ==========  Course Name: Data Structures  Course Code: CSE220  Course Credit: 3  ========== 3 ==========  Course Name: Programming Language II  Course Code: CSE111  Course Credit: 3 |

Create a **Dog** class so that the tester code generates the given output:

| **Driver Code** | **Expected Output** |
| --- | --- |
| public class TesterDog{  public static void main (String[] args) {  Dog scooby = new Dog();  Dog oldie = new Dog();  Dog goofy = new Dog();    scooby.changeName("Scooby");  goofy.changeName("Goofy");    System.*out*.println("1. ===============");  System.*out*.println(scooby.bark());  System.*out*.println("2. ===============");  System.*out*.println(oldie.bark());  System.*out*.println("3. ===============");  oldie.changeColor("White");  System.*out*.println("4. ===============");  System.*out*.println(oldie.bark());  System.*out*.println("5. ===============");  System.*out*.println(goofy.bark());  System.*out*.println("6. ===============");  scooby.changeColor("Brown");  System.*out*.println("7. ===============");  System.*out*.println(scooby.bark());  System.*out*.println("8. ===============");  goofy.changeColor("Black");  }  } | 1. ===============  Scooby is barking  2. ===============  A dog is barking  3. ===============  This dog is White  4. ===============  White dog is barking  5. ===============  Goofy is barking  6. ===============  Scooby is Brown  7. ===============  Scooby the Brown dog is barking  8. ===============  Goofy is Black |

Design the **Reader** class in such a way so that the following code provides the expected output.

* A reader will have a name, capacity to read and an array of books they are reading.
* The initial capacity of a reader will be 0. The initial name will be “New user”.
* A new array is created every time a reader’s capacity is increased, which replaces the initial array.

| **Driver Code** | **Expected Output** |
| --- | --- |
| public class Reader\_tester {  public static void main(String[] args){  Reader r1 = new Reader();  Reader r2 = new Reader();    r1.createReader("Albert", 2);  r2.createReader("Issac", 5);    System.*out*.println("1 ==========");  r1.readerInfo();    System.*out*.println("2 ==========");  r2.addBook("Java");  r2.addBook("Python");  r2.addBook("C++");  r2.readerInfo();    System.*out*.println("3 ==========");  r1.addBook("C#");  r1.addBook("Rust");  r1.addBook("GoLang");    System.*out*.println("4 ==========");  r1.increaseCapacity(5);  r1.addBook("Python");    System.*out*.println("5 ==========");  r1.readerInfo();  }  } | 1 ==========  Name: Albert  Capacity: 2  Books:  No books added yet  2 ==========  Name: Issac  Capacity: 5  Books:  Book 1: Java  Book 2: Python  Book 3: C++  3 ==========  No more space for new book  4 ==========  Albert's capacity increased to 5  5 ==========  Name: Messi  Capacity: 5  Books:  Book 1: C#  Book 2: Rust  Book 3: Python |

| **1** | public class Task11 { |
| --- | --- |
| **2** | public int p = 3, y = 2, sum; |
| **3** | public void methodA(){ |
| **4** | int x = 0, y = 0; |
| **5** | y = y + this.y; |
| **6** | x = sum + 2 + p; |
| **7** | sum = x + y + methodB(p, y); |
| **8** | System.*out*.println(x + " " + y+ " " + sum); |
| **9** | } |
| **10** | public int methodB(int p, int n){ |
| **11** | int x = 0; |
| **12** | y = y + (++p); |
| **13** | x = x + 2 + n; |
| **14** | sum = sum + x + y; |
| **15** | System.*out*.println(x + " " + y+ " " + sum); |
| **16** | return sum; |
| **17** | } |
| **18** | } |

**Driver code:**

| public class Tester11 {  public static void main(String [] args){  Task11 t1 = new Task11 ();  t1.methodA();  t1.methodA();  Task11 t2 = new Task11();  System.out.println(t2.methodB(2,3));  }  } | **Outputs** | | |
| --- | --- | --- | --- |
| **x** | **y** | **Sum** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

| 1 | public class Test2 { |
| --- | --- |
| 2 | int x = 3, y = 1, z = -4; |
| 3 | double p = 2.5; |
| 4 | public void methodA(int n, int x) { |
| 5 | this.x = methodB(x, n); |
| 6 | p = this.x + n % x \* 2.0; |
| 7 | y = (z++) + methodB(z, (int) p) + (++z); |
| 8 | System.out.println(this.x + " " + (n + y) + " " + (x + z)) ; |
| 9 | } |
| 10 | public int methodB(int q, int n) { |
| 11 | int arr[] = {2, -5, 6}; |
| 12 | arr[0] = arr[2] - this.x + n; |
| 13 | arr[1] = q - arr[1]; |
| 14 | arr[2] = arr[q % 3] + arr[n % 2]; |
| 15 | System.out.println(arr[0] + " " + arr[1] + " " + arr[2]) ; |
| 16 | return arr[1] + arr[2] - arr[0]; |
| 17 | } |
| 18 | } |

| public class Tester2{  public static void main(String [] args){  Test2 t = new Test2();  t.methodA(3, 4);  }  } | **Outputs** | | |
| --- | --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |