**Adhish Bahl**

**2347203**

**2MCA B**

**Java**

**Lab 1**

**Code:**

// Implement the concept of Class, Data members, Methods, Access Specifier,  Default Constructor, Method Overloading (minimum 3 methods), Constructor overloading (minimum of 2) in your selected domain.

class SneakersStore {

    private String brand;

    private String name;

    private float price;

    private int quantity;

    SneakersStore() {

        brand = "Not Defined";

        name = "Not Defined";

        price = 0.0f;

        quantity = 0;

    }

    SneakersStore(String brand, String name, float price, int quantity) {

        this.brand = brand;

        this.name = name;

        this.price = price;

        this.quantity = quantity;

    }

    public void setInfo(String newBrand, String newName, float newPrice, int newQuantity) {

        this.brand = newBrand;

        this.name = newName;

        this.price = newPrice;

        this.quantity = newQuantity;

    }

    public void setInfo(String newBrand, String newName) {

        this.brand = newBrand;

        this.name = newName;

    }

    public void setInfo(String newBrand, String newName, float newPrice) {

        this.brand = newBrand;

        this.name = newName;

        this.price = newPrice;

    }

    public void setInfo(float newPrice) {

        this.price = newPrice;

    }

    public void setInfo(int newQuantity) {

        this.quantity = newQuantity;

    }

    public void setInfo(float newPrice, int newQuantity){

        this.price = newPrice;

        this.quantity = newQuantity;

    }

    public void displaySneaker() {

        System.out.println("\nBrand: " + brand);

        System.out.println("Model: " + name);

        System.out.println("Price: Rs. " + price);

        System.out.println("Quantity: " + quantity);

    }

    public void getBrand() {

        System.out.println("Brand: " + this.brand);

    }

    public void getName() {

        System.out.println("Name: " + this.name);

    }

    public void getPrice() {

        System.out.println("Price: " + this.price);

    }

    public void getQuantity() {

        System.out.println("Quantity: " + this.quantity);

    }

}

class SneakerStore {

    public static void main(String[] args) {

        System.out.println("\nSneaker 1: ");

        SneakersStore sneaker1 = new SneakersStore("Nike", "Air Max 90", 15000.00f, 23);

        sneaker1.displaySneaker();

        System.out.println("\nAfter changing name of the Sneaker: ");

        sneaker1.setInfo("Nike", "Running Pro Max");

        sneaker1.getName();

        System.out.println("===============================================================\n");

        System.out.println("\nSneaker 2: ");

        SneakersStore sneaker2 = new SneakersStore("Adidas", "Superstar", 82000.00f, 11);

        sneaker2.displaySneaker();

        System.out.println("\nAfter updating the price of the Sneaker: ");

        sneaker2.setInfo(24000.00f);

        sneaker2.getPrice();

        System.out.println("===============================================================\n");

        SneakersStore sneaker3 = new SneakersStore();

        SneakersStore sneaker4 = new SneakersStore();

        System.out.println("\nSneaker 3: ");

        sneaker3.setInfo("New Balance", "Rocknew 3.0", 32504.35f, 20);

        sneaker3.displaySneaker();

        System.out.println("\nAfter updating the quantity of the Sneaker: ");

        sneaker3.setInfo(4);

        sneaker3.getQuantity();

        System.out.println("===============================================================\n");

        System.out.println("\nSneaker 4: ");

        sneaker4.setInfo("Reebok", "New Look 2.0");

        System.out.println("\n");

        sneaker4.getPrice();

        sneaker4.getQuantity();

        System.out.println("\nAfter assigning the price and the quantity of the Sneaker: ");

        sneaker4.setInfo(1215.32f, 10);

        sneaker4.displaySneaker();

        System.out.println("===============================================================\n");

    }

}

**Output:**



