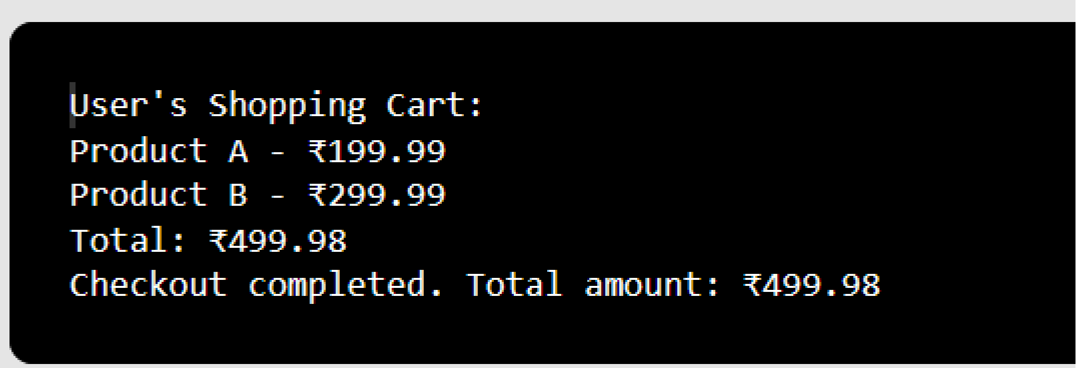
Real-life scenario-based questions related to the Java Collection Framework:

Objective:
These scenarios demonstrate how to utilize Java's Collection Framework to manage and store data in real-life situations such as an online shopping cart system and a social media platform for posts and comments.

Scenario 1: Online Shopping Cart
Question:
You are designing an online shopping cart system. Each user can add products to their cart and make purchases. How would you implement the cart system using Java's Collection Framework?

```
import java.util.ArrayList;
import java.util.List;

class Product {
    private int productId;
    private String name;
    private double price;

    public Product(int productId, String name, double price) {
        this.productId = productId;
        this.name = name;
        this.price = price;
    }

    public double getPrice() {
        return price;
    }

    @Override
    public String toString() {
        return name + " - ₹" + price;
    }
}

class ShoppingCart {
    private List<Product> items = new ArrayList<>();

    public void addItem(Product product) {
        items.add(product);
    }
    public void removeItem(Product product) {
        items.remove(product);
    }

    public List<Product> getItems() {
        return items;
    }

    public double calculateTotal() {
        double total = 0;
        for (Product item : items) {
            total += item.getPrice();
        }
        return total;
    }

    public void checkout() {
        System.out.println("Checkout completed. Total amount: ₹" + calculateTotal());
    }
}

class User {
    private String userId;
    private ShoppingCart cart;

    public User(String userId) {
        this.userId = userId;
        this.cart = new ShoppingCart();
    }

    public ShoppingCart getCart() {
        return cart;
    }
}

public class OnlineShoppingApp {
    public static void main(String[] args) {
        User user1 = new User("user123");
        Product product1 = new Product(1, "Product A", 199.99);
        Product product2 = new Product(2, "Product B", 299.99);

        user1.getCart().addItem(product1);
        user1.getCart().addItem(product2);
        System.out.println("User's Shopping Cart:");
        for (Product item : user1.getCart().getItems()) {
            System.out.println(item);
        }

        double total = user1.getCart().calculateTotal();
        System.out.println("Total: ₹" + total);

        // Simulate the checkout process
        user1.getCart().checkout();
    }
}
```



```
User's Shopping Cart:
Product A - ₹199.99
Product B - ₹299.99
Total: ₹499.98
Checkout completed. Total amount: ₹499.98
```

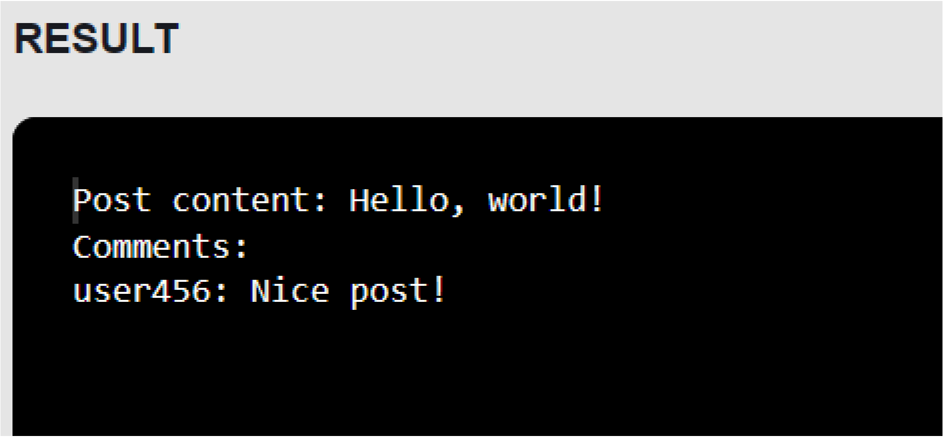
Scenario 2: Social Media Posts and Comments

Question:

Design a system to manage social media posts and their comments. How would you store and retrieve this information efficiently using Java's Collection Framework?

```
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

class User {
    private String userId;
    // Other user-related properties and methods

    public User(String userId) {
        this.userId = userId;
    }

    public String getUserId() {
        return userId;
    }
}

class Comment {
    private int commentId;
    private User author;
    private String content;

    public Comment(int commentId, User author, String content) {
        this.commentId = commentId;
        this.author = author;
        this.content = content;
    }

    public User getAuthor() {
        return author;
    }

    public String getContent() {
        return content;
    }
}

class Post {
    private int postId;
    private User author;
    private String content;
    private List<Comment> comments;

    public Post(int postId, User author, String content) {
        this.postId = postId;
        this.author = author;
        this.content = content;
        this.comments = new ArrayList<>();
    }

    public void addComment(Comment comment) {
        comments.add(comment);
    }

    public List<Comment> getComments() {
        return comments;
    }
    public int getPostId() {
        return postId;
    }

    public String getContent() {
        return content;
    }
}

class SocialMediaSystem {
    private Map<Integer, Post> posts;

    public SocialMediaSystem() {
        this.posts = new HashMap<>();
    }

    public void addPost(Post post) {
        posts.put(post.getPostId(), post);
    }

    public Post getPost(int postId) {
        return posts.get(postId);
    }
}

public class SocialMediaApp {
    public static void main(String[] args) {
        User user1 = new User("user123");
        User user2 = new User("user456");

        Post post1 = new Post(1, user1, "Hello, world!");
        Comment comment1 = new Comment(1, user2, "Nice post!");
        post1.addComment(comment1);

        SocialMediaSystem socialMedia = new SocialMediaSystem();
        socialMedia.addPost(post1);

        // Retrieve a post and its comments
        Post retrievedPost = socialMedia.getPost(1);
        if (retrievedPost != null) {
            System.out.println("Post content: " + retrievedPost.getContent());
            System.out.println("Comments:");
            for (Comment comment : retrievedPost.getComments()) {
                System.out.println(comment.getAuthor().getUserId() + ": " + comment.getContent());
```

```
      }
    }
  }
}
```



```
Post content: Hello, world!
Comments:
user456: Nice post!
```