

Lesson 05 Demo 02

Creating ArrayList

Objective: To implement the use of ArrayList in Java for managing the shopping cart functionality in an e-commerce application

Tools: Eclipse IDE

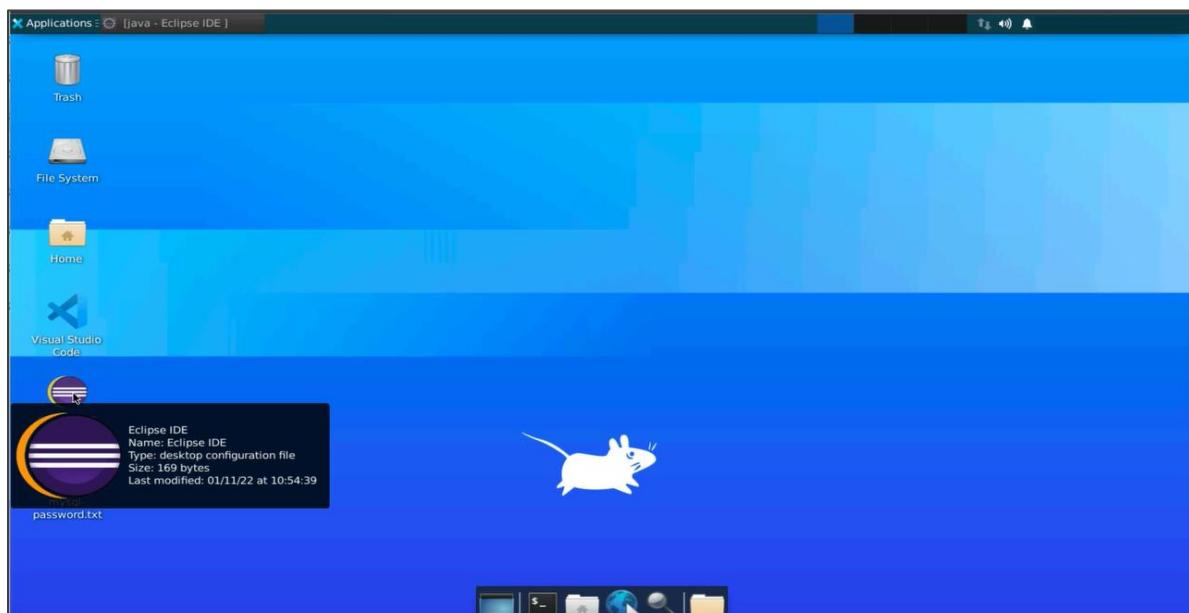
Prerequisites: None

Steps to be followed:

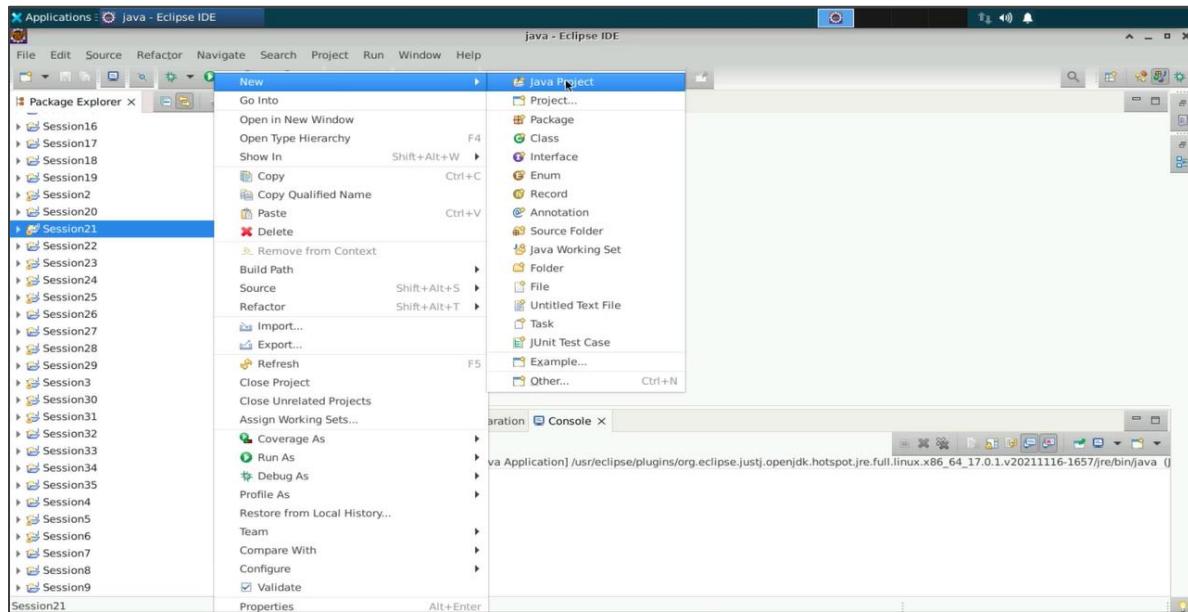
1. Open Eclipse IDE and create a new class
2. Add code for shopping cart and run the project

Step 1: Open Eclipse IDE and create a new class

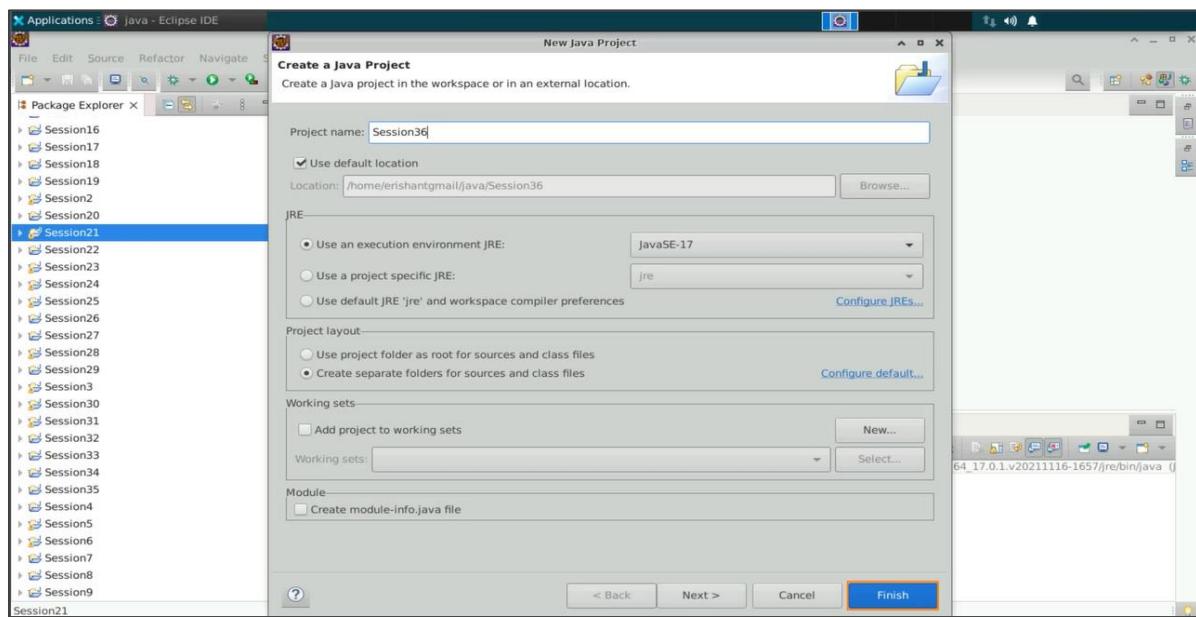
1.1 Open the Eclipse IDE



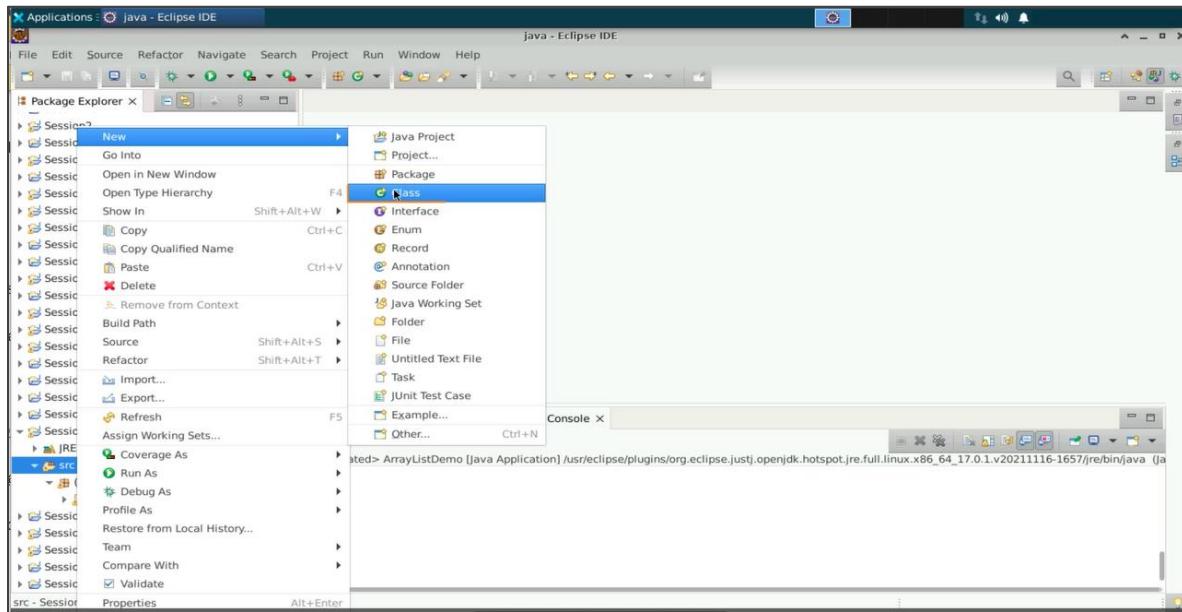
1.2 Right-click, select New, and create a new java project



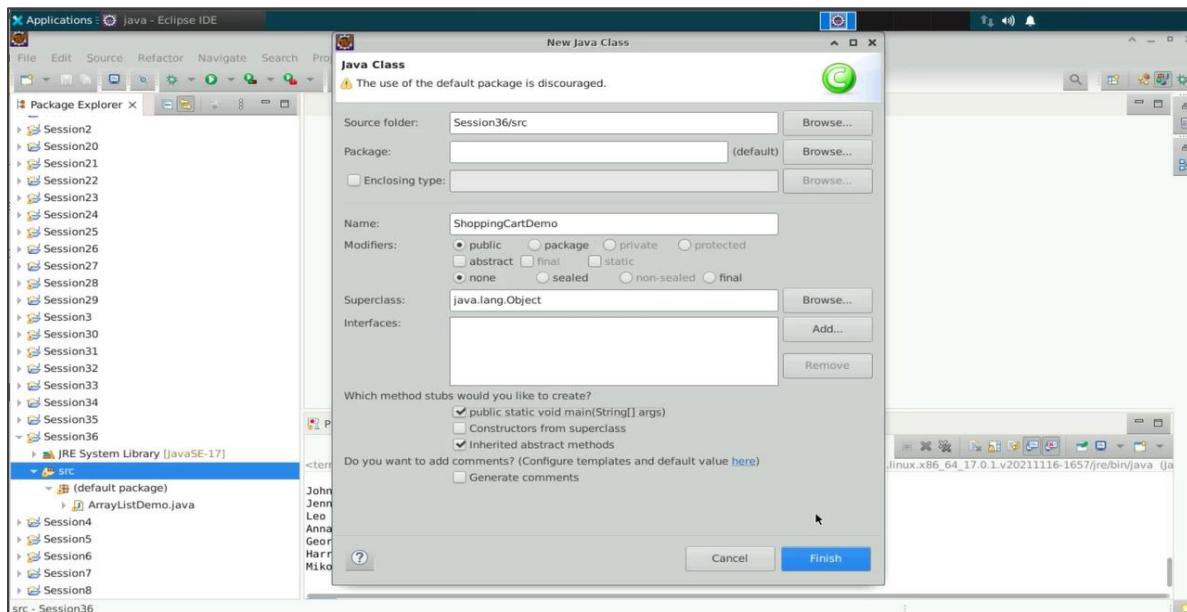
1.3 Name the new Java project Session36



1.4 Select **src**, click on **New**, and then select **Class** option



1.5 Name it **ShoppingCartDemo**, check the main string method, and click the **Finish** button



Step 2: Add code for shopping cart and run the project

2.1 Add the code given below in **ShoppingCartDemo.java** and run the project as a Java program

```
import java.util.ArrayList;

class Product{

    int pid;
    String name;
    int price;
    int quantity;

    public Product() {
        // TODO Auto-generated constructor stub
    }

    public Product(int pid, String name, int price, int quantity) {
        this.pid = pid;
        this.name = name;
        this.price = price;
        this.quantity = quantity;
    }

    @Override
    public String toString() {
        return "Product [pid=" + pid + ", name=" + name + ", price=" + price + ",
quantity=" + quantity + "]";
    }
}

class User{
```

```
String name;  
String email;  
  
ArrayList<Product> shoppingCart;  
  
public User() {  
    // TODO Auto-generated constructor stub  
}  
  
  
public User(String name, String email) {  
    this.name = name;  
    this.email = email;  
    shoppingCart = new ArrayList<Product>();  
}  
  
  
void addProductToCart(Product product){  
    product.quantity++;  
    shoppingCart.add(product);  
}  
  
  
void increment(Product product) {  
    int idx = shoppingCart.indexOf(product);  
    if(idx>0)  
        shoppingCart.get(idx).quantity++;  
  
}  
  
  
void decrement() {  
    // write decrement for quantity here  
}  
  
  
void removeProductFromCart(Product product) {  
    int idx = shoppingCart.indexOf(product);  
    shoppingCart.remove(idx);  
}  
  
  
void showProductsInCart() {
```

```
System.out.println("Listing Products for "+name);

for(Product product : shoppingCart) {
    System.out.println(product);
}

System.out.println();
}

void checkOutFromCart() {

    System.out.println("Checking Out for "+name);

    int total = 0;

    for(Product product : shoppingCart) {
        System.out.println(product);
        total += (product.price*product.quantity);
    }

    System.out.println("Dear, "+name+" Please Pay: "+total);
    System.out.println();
}

}

public class ShoppingCartDemo {

    public static void main(String[] args) {

        Product p1 = new Product(101, "Adidas AlphaBounce Shoe", 2000, 0);
        Product p2 = new Product(201, "iPhone 11", 60000, 0);
        Product p3 = new Product(301, "Samsung LED TV", 5000, 0);
        Product p4 = new Product(401, "Hitachi Fridge", 80000, 0);
        Product p5 = new Product(501, "Coffee Mug", 200, 0);
    }
}
```

```
User user1 = new User("John", "john@example.com");
User user2 = new User("Fionna", "fionna@example.com");

user1.addProductToCart(p1);
user1.addProductToCart(p2);
user1.addProductToCart(p3);

user2.addProductToCart(p4);
user2.addProductToCart(p5);

user1.removeProductFromCart(p2);

user1.increment(p1);

user2.increment(p5);
user2.increment(p5);

user1.showProductsInCart();

user2.showProductsInCart();

user1.checkOutFromCart();
user2.checkOutFromCart();
}

}
```

The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** Applications > java - Session36/src/Sho...
- Menu Bar:** File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help
- Toolbar:** Standard Java development toolbar.
- Left Panel (Project Explorer):** Shows ShoppingCartDemo.java as the active file.
- Code Editor:** Displays the ShoppingCartDemo.java code. The code creates products and users, adds products to their carts, removes one, increments quantities, and shows the cart contents for each user before checking them out.
- Output View:** Shows the terminal output of the application's execution. It lists products (Adidas AlphaBounce Shoe, iPhone 11, Samsung LED TV, Hitachi Fridge, Coffee Mug) and users (John, Fiona). It tracks the quantity of each item in the cart and provides a total price for Fiona's cart.

```
java - Session36/src/ShoppingCartDemo.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

ShoppingCartDemo.java X
Problems Javadoc Declaration Console X

97 public class ShoppingCartDemo {
98
99     public static void main(String[] args) {
100
101         Product p1 = new Product(101, "Adidas AlphaBounce Shoe", 2000, 0);
102         Product p2 = new Product(201, "iPhone 11", 60000, 0);
103         Product p3 = new Product(301, "Samsung LED TV", 5000, 0);
104         Product p4 = new Product(401, "Hitachi Fridge", 80000, 0);
105         Product p5 = new Product(501, "Coffee Mug", 200, 0);
106
107         User user1 = new User("John", "john@example.com");
108         User user2 = new User("Fionna", "fionna@example.com");
109
110         user1.addProductToCart(p1);
111         user1.addProductToCart(p2);
112         user1.addProductToCart(p3);
113
114         user2.addProductToCart(p4);
115         user2.addProductToCart(p5);
116
117         user1.removeProductFromCart(p2);
118
119         user1.increment(p1);
120
121         user2.increment(p5);
122         user2.increment(p5);
123
124         user1.showProductsInCart();
125
126         user2.showProductsInCart();
127
128         user1.checkOutFromCart();
129         user2.checkOutFromCart();
130
131
132
133
134     }
}

<terminated> ShoppingCartDemo [Java Application] /usr/eclipse/plugins/org.eclipse.justj/or
Product [pid=101, name=Adidas AlphaBounce Shoe, price=2000, quantity=1]
Product [pid=301, name=Samsung LED TV, price=5000, quantity=1]

Listing Products for Fionna
Product [pid=401, name=Hitachi Fridge, price=80000, quantity=1]
Product [pid=501, name=Coffee Mug, price=200, quantity=3]

Checking Out for John
Product [pid=101, name=Adidas AlphaBounce Shoe, price=2000, quantity=1]
Product [pid=301, name=Samsung LED TV, price=5000, quantity=1]
Dear, John Please Pay: 7000 I

Checking Out for Fionna
Product [pid=401, name=Hitachi Fridge, price=80000, quantity=1]
Product [pid=501, name=Coffee Mug, price=200, quantity=3]
Dear, Fionna Please Pay: 80600
```

By following these steps, you will gain hands-on experience in creating classes, managing collections of objects using ArrayLists, and implementing basic operations.