

UNIVERSIDAD DE LAS FUERZAS ARMADAS

ESPE



Object-oriented programming

NRC: 9642

Team 2

Topic: ReviewSolidPrinciples

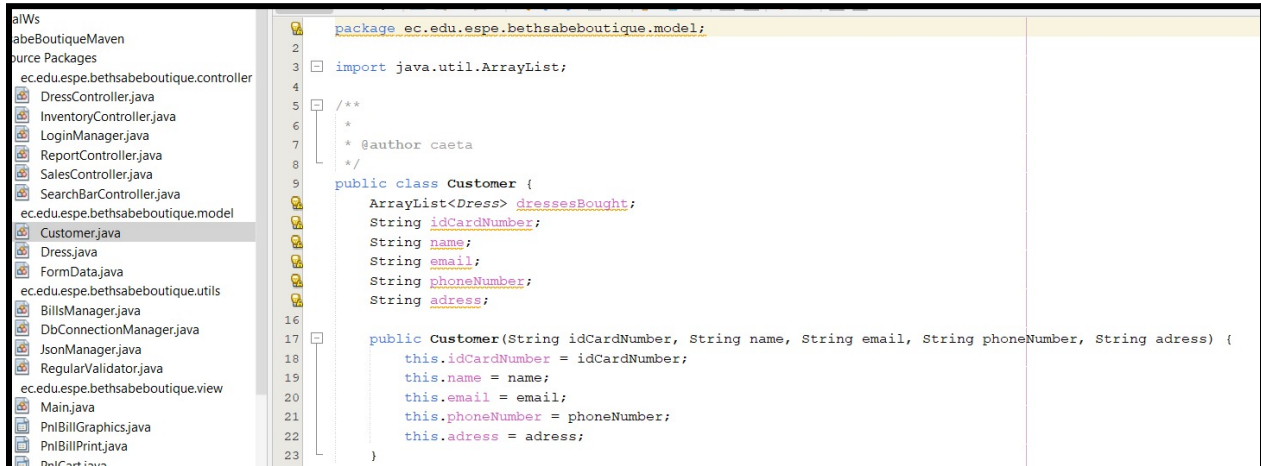
CHIPE ZAMBRANO PAMELA NAOMI

COBEÑA ZAMBRANO JOAN OSWALDO

DAVILA MOLINA ANABEL

ESPIN ANDRADE ANDRES ALEXANDER

Review Solid Principles Team 2 to project Team 3



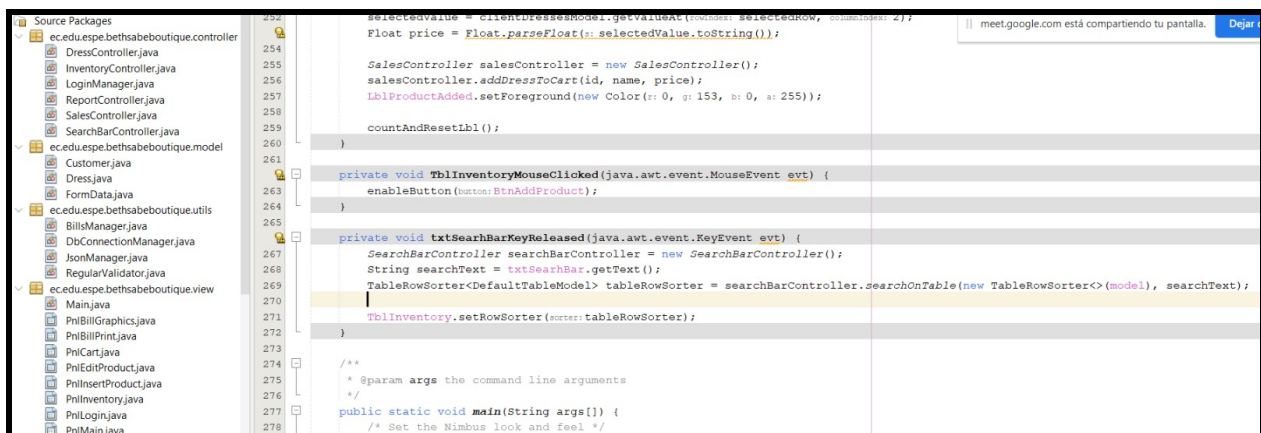
```
package ec.edu.espe.bethsabeboutique.model;

import java.util.ArrayList;

/**
 * @author caeta
 */
public class Customer {
    ArrayList<Dress> dressesBought;
    String idCardNumber;
    String name;
    String email;
    String phoneNumber;
    String address;

    public Customer(String idCardNumber, String name, String email, String phoneNumber, String address) {
        this.idCardNumber = idCardNumber;
        this.name = name;
        this.email = email;
        this.phoneNumber = phoneNumber;
        this.address = address;
    }
}
```

no get and set y no private



```
selectedValue = ClientDressesModel.getValueAt(selectedRow, columnCount - 2);
Float price = Float.parseFloat(selectedValue.toString());

SalesController salesController = new SalesController();
salesController.addDressToCart(id, name, price);
LblProductAdded.setForeground(new Color(r: 0, g: 153, b: 0, a: 255));

countAndResetLbl();

private void tblInventoryMouseClicked(java.awt.event.MouseEvent evt) {
    enableButton(button: BtnAddProduct);
}

private void txtSearchBarKeyReleased(java.awt.event.KeyEvent evt) {
    SearchBarController searchBarController = new SearchBarController();
    String searchText = txtSearchBar.getText();
    TableRowSorter<DefaultTableModel> tableRowSorter = searchBarController.searchOnTable(new TableRowSorter<>(model), searchText);
    tblInventory.setRowSorter(sorter: tableRowSorter);
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    // Set the Nimbus look and feel
}
```

s aplicado



```
public DressController() {
    this.client = DbConnectionManager.getInstance().getClient();
    inventoryController = new InventoryController();
}

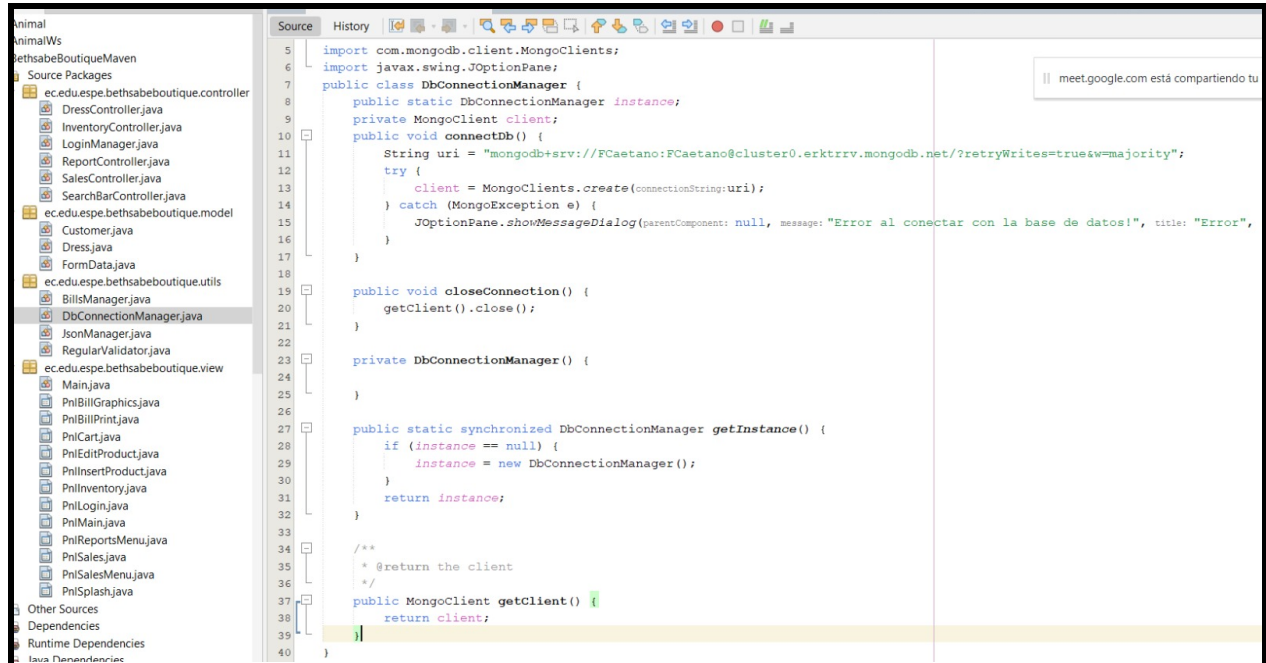
public String convertFormToDress(FormData formData) {
    int id = calculateId();
    float price = Float.parseFloat(formData.getPrice());
    String name = formData.getName();
    String size = formData.getSize();
    String registerDate = formData.getActualDate();
    int quantity = Integer.parseInt(formData.getQuantity());

    Dress dress = new Dress(id, price, name, size, registerDate, quantity);
    String dressData = jsonManager.convertToJson(dress);

    return dressData;
}

public int calculateId() {
    MongoCollection collection = client.getDatabase(string: "Bethsabeboutique").getCollection(string: "Aux");
    Document filter = new Document(); // Filtro vacio para obtener todos los documentos
    FindIterable<Document> documents = collection.find(bson: filter);
}
```

Dependency



Static methods



Singleton