**📌 import Statements**

import javax.swing.\*;

import java.awt.\*;

import java.io.FileWriter;

import java.io.IOException;

* javax.swing.\*: GUI তৈরির জন্য প্রয়োজনীয় ক্লাসগুলো ইমপোর্ট করে (যেমন JFrame, JComboBox, JButton ইত্যাদি)।
* java.awt.\*: Layout এবং Component design-এর জন্য ব্যবহৃত হয়।
* java.io.FileWriter: ফাইল লেখার জন্য।
* java.io.IOException: ফাইল লেখার সময় কোনো Input/Output exception ধরার জন্য।

**📌 FoodDeliveryApp ক্লাস ডেফিনিশন**

public class FoodDeliveryApp extends JFrame {

এই ক্লাসটি JFrame (একটি উইন্ডো) থেকে extends করা হয়েছে, মানে এটি একটি GUI উইন্ডো হিসেবে কাজ করবে।

**📌 Variable Declaration**

private JComboBox<String> drinksMenu, burgerMenu, biryaniMenu, snacksMenu, pizzaMenu;

private JTextField customerNameField, addressField, phoneField;

private JTextArea orderSummary;

private JButton orderButton, clearButton;

private JLabel totalLabel;

private double totalPrice;

এখানে বিভিন্ন GUI উপাদান ডিক্লেয়ার করা হয়েছে, যেমন: ড্রপডাউন মেনু, টেক্সট ফিল্ড, বোতাম, লেবেল ইত্যাদি।

**📌 ফাইল পাথ**

private static final String ORDER\_FILE\_PATH = "C:\\Users\\User\\OneDrive\\Desktop\\food odering apps\\oders.txt";

অর্ডার সংরক্ষণের জন্য একটি নির্দিষ্ট ফাইল পাথ সেট করা হয়েছে।

**📌 Constructor (মূল অংশ)**

public FoodDeliveryApp() {

এটি কনস্ট্রাক্টর—অর্থাৎ যখন অবজেক্ট তৈরি হবে, তখন এই অংশ চালু হবে।

**1. Window Setup:**

setTitle("Food Delivery App");

setSize(650, 650);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setLocationRelativeTo(null);

setLayout(new BorderLayout());

* উইন্ডোর টাইটেল, সাইজ নির্ধারণ করা হয়েছে।
* setDefaultCloseOperation দিলে ক্লোজ বাটনে ক্লিক করলে অ্যাপ্লিকেশন বন্ধ হয়।
* setLayout দিয়ে উইন্ডোর মধ্যে কম্পোনেন্ট কোথায় বসবে তা নির্ধারণ করা হয়।

**2. Combo Boxes (Drop-down Lists):**

drinksMenu = new JComboBox<>(new String[]{

"Select a Drinks & Beverages item", ...

});

এভাবে প্রতিটি বিভাগ (ড্রিঙ্কস, বার্গার, বিরিয়ানি ইত্যাদি)-এর জন্য একটি ড্রপডাউন মেনু তৈরি করা হয়েছে।

**3. Text Fields and Summary Area:**

customerNameField = new JTextField(12);

addressField = new JTextField(15);

phoneField = new JTextField(10);

orderSummary = new JTextArea(12, 50);

* গ্রাহকের নাম, ঠিকানা, ফোন নম্বর ইনপুট দেওয়ার জন্য JTextField।
* অর্ডার রিভিউ দেখানোর জন্য JTextArea।

**4. Buttons and Label:**

orderButton = new JButton("Place Order");

clearButton = new JButton("Clear Order");

totalLabel = new JLabel("Total: ৳0.00 BDT");

* একটি অর্ডার প্লেস করার বোতাম ও একটি ক্লিয়ার বোতাম।
* মোট দাম দেখানোর জন্য একটি লেবেল।

**5. Layout Design (GridLayout):**

JPanel inputPanel = new JPanel(new GridLayout(9, 2, 5, 5));

inputPanel.add(...);

* JPanel ব্যবহার করে ইনপুট এলিমেন্টগুলো ৯x২ গ্রিডে রাখা হয়েছে (৯টি লেবেল + ইনপুট)।

**📌 অর্ডার বোতামের কাজ**

orderButton.addActionListener(e -> {

* বোতামে ক্লিক করলে নিচের কাজগুলো হবে:

1. ফর্ম চেক করবে সব ইনপুট দেওয়া আছে কিনা।
2. প্রতিটি খাবার মেনু থেকে বাছাই করা আইটেম নেবে।
3. addSelectedItem() মেথড কল করে অর্ডার ডিটেইলসে যোগ করবে।
4. মোট দাম হিসাব করবে।
5. স্ক্রিনে দেখাবে এবং ফাইলে সংরক্ষণ করবে।

**📌 addSelectedItem() Method:**

private boolean addSelectedItem(JComboBox<String> menu, String category, StringBuilder orderDetails) {

* বাছাই করা আইটেমের দাম বের করে মোট দামে যোগ করে।
* StringBuilder-এ অর্ডারের তথ্য যোগ করে।

**📌 getItemPrice() Method:**

private double getItemPrice(String item) throws NumberFormatException {

* ড্রপডাউন থেকে পাওয়া লেখা থেকে দাম বের করে double রূপে রিটার্ন করে।

**📌 clearButton-এর কাজ**

clearButton.addActionListener(e -> {

* ফর্মের সব তথ্য মুছে ফেলে, আবার শুরু করার জন্য।

**📌 main() Method**

public static void main(String[] args) {

SwingUtilities.invokeLater(() -> new FoodDeliveryApp().setVisible(true));

}

* অ্যাপ্লিকেশন চালু করার জন্য main() ফাংশন।
* GUI তৈরি এবং দৃশ্যমান করার জন্য SwingUtilities.invokeLater() ব্যবহার করা হয়েছে।