**Food Delivery App Code Explain:**

**import javax.swing.\* :**

**import javax.swing.\*;** এই লাইনটি জাভা প্রোগ্রামে ব্যবহার করা হয় **Swing লাইব্রেরির সব ক্লাস** ইম্পোর্ট করার জন্য।

**🔹 javax.swing কী?**

javax.swing হলো জাভার একটি **GUI (Graphical User Interface)** লাইব্রেরি। এর মাধ্যমে আপনি উইন্ডো, বোতাম, টেক্সট বক্স, লেবেল, ডায়ালগ বক্স ইত্যাদি **গ্রাফিক্যাল উপাদান (component)** তৈরি করতে পারেন।

উদাহরণ:

* JFrame – একটি উইন্ডো তৈরি করতে
* JButton – একটি বোতাম তৈরি করতে
* JLabel – একটি টেক্সট লেবেল দেখাতে
* JTextField – ইউজার ইনপুট নেয়ার জন্য টেক্সট ফিল্ড

**🔹 import javax.swing.\*; মানে কী?**

এই লাইনে \* ব্যবহার করার অর্থ হচ্ছে javax.swing প্যাকেজের **সকল ক্লাস ও ইন্টারফেস** ইম্পোর্ট করা। যেমন:

import javax.swing.\*;

এর ফলে আপনি আলাদা আলাদা করে নিচের মতো করে লেখার দরকার পড়বে না:

java

CopyEdit

import javax.swing.JFrame;

import javax.swing.JButton;

import javax.swing.JLabel;

**🔹 ব্যবহার কেমন হয়?**

উদাহরণ:

import javax.swing.\*;

public class MyWindow {

public static void main(String[] args) {

JFrame frame = new JFrame("আমার উইন্ডো");

JButton button = new JButton("ক্লিক করুন");

frame.setSize(300, 200);

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.add(button);

frame.setVisible(true);

}

}

এই কোডে আমরা JFrame ও JButton ব্যবহার করেছি, যেগুলো javax.swing প্যাকেজে রয়েছে। তাই আমাদের import javax.swing.\*; লাগছে।

**import java.awt.\*:**

**import java.awt.\*;** — এই লাইনটি Java প্রোগ্রামে awt (Abstract Window Toolkit) লাইব্রেরির সকল ক্লাস এবং ইন্টারফেস ব্যবহার করার জন্য লেখা হয়।

**🔹 java.awt কী?**

java.awt হচ্ছে Java-এর একটি বিল্ট-ইন প্যাকেজ যেটা GUI (Graphical User Interface) তৈরি করার জন্য ব্যবহৃত হয়। এর মাধ্যমে আপনি বিভিন্ন UI উপাদান (যেমন: বাটন, উইন্ডো, টেক্সটফিল্ড, লেবেল ইত্যাদি) তৈরি করতে পারেন।

**🔹 import java.awt.\*; মানে কী?**

এখানে \* চিহ্ন মানে হলো — java.awt প্যাকেজের সব ক্লাস ও ইন্টারফেস ইমপোর্ট করা হয়েছে।

উদাহরণস্বরূপ, নিচের ক্লাসগুলো এই import করার মাধ্যমে ব্যবহার করা যায়:

* Frame
* Button
* Label
* TextField
* Color
* Graphics
* ইত্যাদি।

**🔹 উদাহরণ:**

import java.awt.\*;

public class MyWindow {

public static void main(String[] args) {

Frame f = new Frame("আমার প্রথম উইন্ডো");

f.setSize(300, 200);

f.setVisible(true);

}

}

🔸 এখানে Frame ক্লাসটি এসেছে java.awt প্যাকেজ থেকে।

**import java.io.FileWriter;:**

Java-তে **import java.io.FileWriter;** লাইনের মাধ্যমে আমরা FileWriter ক্লাসটিকে আমাদের প্রোগ্রামে ব্যবহার করার জন্য ইমপোর্ট করি। এটা java.io প্যাকেজের অংশ, এবং মূলত ফাইলের মধ্যে লেখার (writing to file) জন্য ব্যবহৃত হয়।

### import java.io.FileWriter; এর বাংলা ব্যাখ্যা:

"আমি java.io প্যাকেজের মধ্যে থাকা FileWriter নামের ক্লাসটি আমার প্রোগ্রামে ব্যবহার করতে চাই।"

FileWriter হলো একটি Java ক্লাস, যেটি ব্যবহার করে আপনি টেক্সট বা স্ট্রিং কোনো ফাইলে লিখে রাখতে পারেন।

### উদাহরণসহ ব্যাখ্যা:

import java.io.FileWriter;

import java.io.IOException;

public class Main {

public static void main(String[] args) {

try {

FileWriter writer = new FileWriter("output.txt");

writer.write("হ্যালো, এটি একটি ফাইল লেখার উদাহরণ।");

writer.close();

System.out.println("সফলভাবে ফাইলে লেখা হয়েছে।");

} catch (IOException e) {

System.out.println("একটি সমস্যা হয়েছে: " + e.getMessage());

}

}

}

#### এখানে যা হচ্ছে:

1. FileWriter writer = new FileWriter("output.txt"); – এটি একটি output.txt নামের ফাইল তৈরি করছে (না থাকলে তৈরি হবে), অথবা থাকলে সেটিকে ওপেন করছে লেখার জন্য।
2. writer.write(...) – এটি ফাইলের মধ্যে কিছু টেক্সট লিখছে।
3. writer.close() – লেখার পর ফাইলটি বন্ধ করে দিচ্ছে, যাতে সব কিছু সেভ হয়।

**import java.io.IOException;:**

**import java.io.IOException**; এই লাইনটি Java প্রোগ্রামে ব্যবহৃত হয় IOException ক্লাসটি ব্যবহার করার জন্য।

**🔹 import java.io.IOException; মানে কী?**

এই লাইনটি IOException ক্লাসটিকে **Java I/O (Input/Output)** লাইব্রেরি থেকে আমদানি করে। এটি **Checked Exception** অর্থাৎ এমন একটি এক্সসেপশন যা কম্পাইল টাইমেই হ্যান্ডেল করতে হয় (try-catch ব্লকে বা throws দিয়ে)।

**🔹 IOException কবে লাগে?**

যখন আপনি Java-তে ফাইল পড়া, লেখা বা কোনো ইনপুট/আউটপুট অপারেশন করেন (যেমন: FileReader, BufferedReader, InputStream, ইত্যাদি), তখন যদি কোনো সমস্যা হয় (যেমন ফাইল না পাওয়া যায়), তখন IOException ছোড়া হয়।

**🔹 উদাহরণ:**

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;

public class Example {

public static void main(String[] args) {

try {

BufferedReader reader = new BufferedReader(new FileReader("data.txt"));

String line = reader.readLine();

System.out.println(line);

reader.close();

} catch (IOException e) {

System.out.println("File পড়তে সমস্যা হয়েছে: " + e.getMessage());

}

}

}

**এই কোডে:**

* data.txt ফাইলটি পড়তে গিয়ে যদি কোনো সমস্যা হয় (যেমন ফাইল না থাকে), তাহলে IOException হবে।
* তাই IOException হ্যান্ডেল করার জন্য catch ব্লকে কোড লেখা হয়েছে।

**✅ সংক্ষেপে:**

* IOException হলো I/O সংক্রান্ত সমস্যা ধরার জন্য একটি Exception।
* একে ব্যবহার করতে হলে import java.io.IOException; লিখতে হয়।
* এটি সাধারণত ফাইল ইনপুট/আউটপুট অপারেশনের সময় ব্যবহার হয়।

**public class FoodDeliveryApp extends JFrame {:**

* **public:** এটি একটি **access modifier**, যার মানে হচ্ছে এই ক্লাসটি **সবার জন্য উন্মুক্ত**। অন্য যেকোনো ক্লাস বা প্যাকেজ থেকেও এই ক্লাসটি ব্যবহার করা যাবে।
* **class:** এটি নির্দেশ করে যে আমরা একটি **নতুন ক্লাস** তৈরি করছি। Java-তে একটি ক্লাস হলো একটি ব্লুপ্রিন্ট বা কাঠামো, যা অবজেক্ট তৈরি করতে ব্যবহৃত হয়।
* **FoodDeliveryApp**: এটি হলো **ক্লাসের নাম**। সাধারণভাবে ক্লাসের নাম বড় হাতের অক্ষর দিয়ে শুরু হয় **(naming convention** অনুযায়ী)। এই ক্লাসটি একটি ফুড ডেলিভারি অ্যাপকে প্রতিনিধিত্ব করছে।
* **extends JFrame:** এখানে **extends** শব্দটি বোঝায় যে এই ক্লাসটি **JFrame ক্লাসকে উত্তরাধিকার সূত্রে গ্রহণ করছে**। অর্থাৎ, **FoodDeliveryApp** ক্লাসটি **JFrame**-এর সব বৈশিষ্ট্য ও কার্যক্ষমতা পাবে এবং প্রয়োজনে এগুলোকে পরিবর্তন (**override**) করতেও পারবে।

**সংক্ষেপে বাংলা অনুবাদ:**

“**FoodDeliveryApp** নামে একটি ক্লাস তৈরি করা হয়েছে, যা **JFrame** ক্লাস থেকে বৈশিষ্ট্য উত্তরাধিকার হিসেবে পেয়েছে এবং এটি **public** মানে সবার জন্য উন্মুক্ত।”

**private JComboBox<String> comboBox:**

* **private**: এটি একটি **অ্যাক্সেস মডিফায়ার**। এর মানে হলো এই ভ্যারিয়েবলটি কেবলমাত্র ওই ক্লাসের মধ্যেই ব্যবহার করা যাবে যেখানে এটি ঘোষণা করা হয়েছে। অন্য কোন ক্লাস থেকে সরাসরি অ্যাক্সেস করা যাবে না।
* **JComboBox<String>**:  
  **JComboBox** হলো **Java** **Swing** লাইব্রেরির একটি **GUI (Graphical User Interface) কম্পোনেন্ট**, যা একটি **ড্রপডাউন তালিকা (dropdown list)** তৈরি করে। **<String>** অংশটি বলে দিচ্ছে যে এই ড্রপডাউনে শুধুমাত্র **স্ট্রিং টাইপের** (লেখা বা টেক্সট) আইটেম থাকবে।
* **comboBox**: এটি ভ্যারিয়েবলের নাম। এই ভ্যারিয়েবলটির মাধ্যমে পরবর্তীতে **JComboBox** অবজেক্টকে রেফার করা হবে।

**সারসংক্ষেপে:**

এই লাইনটি একটি প্রাইভেট **JComboBox** ডিক্লেয়ার করছে, যেখানে ব্যবহারকারী স্ট্রিং (টেক্সট) আইটেম থেকে একটি নির্বাচন করতে পারবেন।

**private JTextField:**

এটি Java প্রোগ্রামিং ভাষায় একটি **অপূর্ণ (incomplete) ভ্যারিয়েবল ডিক্লেয়ারেশন**, তবে চলুন .

**🔹 private:**

* এটি একটি **অ্যাক্সেস মডিফায়ার** (Access Modifier)।
* **private** মানে হলো, এই ভ্যারিয়েবলটি **শুধুমাত্র ক্লাসের ভিতরে ব্যবহারযোগ্য**। বাইরে থেকে এক্সেস করা যাবে না।

**🔹 JTextField:**

* এটি হলো **Swing লাইব্রেরির** একটি ক্লাস, যা **টেক্সট ইনপুট ফিল্ড** তৈরিতে ব্যবহার করা হয়।
* ইউজার এখানে কিছু টাইপ করতে পারে, যেমন নাম, সংখ্যা ইত্যাদি।

**সম্পূর্ণ উদাহরণ (বাংলা ব্যাখ্যা সহ):**

**private JTextField nameField;**

🔸 এখানে:

* **nameField** হচ্ছে ভ্যারিয়েবলের নাম।
* এটি **JTextField** টাইপের, অর্থাৎ একটি টেক্সট ইনপুট ফিল্ড।
* এবং **private** হওয়ায়, এটি শুধু ওই ক্লাসের ভিতরে ব্যবহার করা যাবে।

**private JTextArea:**

এই লাইনটি **একটি প্রাইভেট JTextArea টাইপের ভেরিয়েবল** ডিক্লেয়ার করছে। এখানে:

* **private** ➝ এক্সেস মডিফায়ার। এর মানে হলো এই ভেরিয়েবলটি শুধু এই ক্লাসের ভেতর থেকেই ব্যবহার করা যাবে, বাইরের ক্লাস থেকে নয়।
* **JTextArea** ➝ এটি **Swing লাইব্রেরির** একটি ক্লাস, যেটি একটি মাল্টিলাইন টেক্সট ইনপুট এরিয়া তৈরি করতে ব্যবহৃত হয়। অর্থাৎ, ইউজার অনেক লাইন লেখার সুযোগ পায়।
* তবে **এই লাইনে কোনো ভেরিয়েবলের নাম নেই বা অবজেক্ট তৈরি হয়নি**, এটি অসম্পূর্ণ। সঠিকভাবে লিখলে এটি হওয়া উচিত:

**private JTextArea textArea;**

অথবা অবজেক্ট সহ:

**private JTextArea textArea = new JTextArea();**

**🔹 সারাংশে বাংলায়:**

এই লাইনটি একটি **JTextArea** নামের GUI উপাদানকে ঘোষণা করে, যা শুধুমাত্র ক্লাসের ভিতরে ব্যবহারযোগ্য হবে। এটি ইউজার থেকে একাধিক লাইনের লেখা ইনপুট নিতে পারে।

**Private jButton:**

**private JButton** — এই লাইনটি জাভা (Java) প্রোগ্রামিং ভাষায় ব্যবহৃত হয় এবং এটি একটি বোতাম (button) তৈরি করার জন্য ব্যবহৃত একটি কোডের অংশ।

**বাংলায় ব্যাখ্যা:**

* **private**: এটি একটি **অ্যাক্সেস মডিফায়ার** (access modifier)। এর মানে হচ্ছে এই ভ্যারিয়েবল বা অবজেক্টটি কেবলমাত্র একই ক্লাসের ভিতর থেকে অ্যাক্সেস করা যাবে, বাইরের কোথাও থেকে নয়।
* **JButton**: এটি জাভার **Swing লাইব্রেরির** একটি ক্লাস, যা একটি **গ্রাফিক্যাল ইউজার ইন্টারফেস (GUI)** তে বোতাম (Button) তৈরি করতে ব্যবহৃত হয়।

সাধারণত এইভাবে লেখা হয়:

**private JButton myButton;**

বাংলায় পূর্ণ ব্যাখ্যা:

**private JButton myButton;** — এখানে একটি প্রাইভেট **JButton** অবজেক্ট তৈরি করা হয়েছে যার নাম **myButton**, যা কেবলমাত্র সেই ক্লাসের ভিতরে ব্যবহার করা যাবে যেখানে এটি ডিক্লেয়ার করা হয়েছে।

**private JLabel totalLabel;:**

এই লাইনটি একটি **JLabel** নামক অবজেক্ট টাইপের ভ্যারিয়েবল **totalLabel** ঘোষণা করছে। এখানে:

* **private** ➝ এটি একটি **অ্যাক্সেস মডিফায়ার**। এর অর্থ হলো এই ভ্যারিয়েবলটি কেবলমাত্র এই ক্লাসের ভেতরে ব্যবহার করা যাবে। বাইরের কোনো ক্লাস থেকে সরাসরি অ্যাক্সেস করা যাবে না।
* **JLabel** ➝ এটি **Swing লাইব্রেরির** একটি ক্লাস, যা GUI-তে (Graphical User Interface) টেক্সট দেখানোর জন্য ব্যবহৃত হয়।
* **totalLabel** ➝ এটি ভ্যারিয়েবলটির নাম, যার মাধ্যমে JLabel অবজেক্টকে রেফার করা হবে।

**সারাংশে**, এই লাইনটি একটি প্রাইভেট **JLabel** টাইপের ভ্যারিয়েবল ঘোষণা করেছে, যার নাম **totalLabel**, এবং এটি সাধারণত GUI-তে কোনো "total" মান দেখানোর জন্য ব্যবহৃত হতে পারে।

**private double totalPrice:**

**private**: এটি একটি *access modifier*, যা বোঝায় যে এই ভেরিয়েবলটি শুধুমাত্র এই ক্লাসের ভিতরে থেকেই অ্যাক্সেস করা যাবে। বাইরে থেকে সরাসরি ব্যবহার করা যাবে না।

**double**: এটি একটি ডেটা টাইপ, যা দশমিক সংখ্যা (যেমন: 10.5, 99.99 ইত্যাদি) সংরক্ষণ করতে ব্যবহৃত হয়। অর্থাৎ এটি ভগ্নাংশসহ সংখ্যা রাখার জন্য ব্যবহৃত হয়।

**totalPrice**: এটি ভেরিয়েবলের নাম, যার অর্থ এখানে "মোট দাম" বোঝাতে ব্যবহৃত হয়েছে।

**সম্পূর্ণ ব্যাখ্যা (বাংলায়):**

এই লাইনটি একটি প্রাইভেট double টাইপের ভেরিয়েবল ঘোষণা করছে যার নাম totalPrice। এটি একটি দশমিক সংখ্যা রাখবে এবং কেবলমাত্র ক্লাসের ভিতর থেকেই ব্যবহারযোগ্য হবে।