2 industry-oriented assignments on Built in functional interfaces in Java.

Assignment 1: Bank Transaction System

Scenario

A bank system needs to process customer transaction amounts.

Requirements

1. **Predicate<Double>** – Check if a transaction is **suspicious** (greater than ₹50,000).
2. **Consumer<Double>** – Print a message for each transaction:
   "Processed transaction of ₹<amount>".
3. **Supplier<Integer>** – Generate a random **4-digit OTP** for transaction verification.
4. **Function<Double, Double>** – Deduct a **2% processing fee** from each transaction.
5. **BiFunction<Double, Double, Double>** – Apply a **discount on processing fee** if customer has loyalty points:
   finalAmount = amount - (amount * feePercentage / 100).

Input Example

List<Double> transactions = Arrays.asList(1200.0, 55000.0, 30000.0);

Expected Flow

- Predicate → Detects ₹55,000 as suspicious.
- Consumer → Prints: “Processed transaction of ₹1200.0” etc.
- Supplier → Generates OTP: Transaction OTP: 4821.
- Function → Deducts processing fee (2%).
- BiFunction → Applies discount on fee if loyalty points exist.

👉 **Learning Outcome:** Using **all functional interfaces together** to simulate **banking workflows** with only wrapper types.

🚀 **Assignment 2: Online Shopping System**

Scenario

An e-commerce platform processes order amounts.

Requirements

1. **Predicate<Double>** – Check if order qualifies for **free shipping** (above ₹2000).
2. **Consumer<Double>** – Print the order confirmation:
   "Order placed successfully for ₹<amount>".
3. **Supplier<String>** – Generate a random **coupon code** (like "SAVE123").

4. **Function<Double, Double>** – Convert price from **USD to INR** (1 USD = 83 INR).
5. **BiFunction<Double, Double, Double>** – Apply **percentage discount** on an order: finalPrice = price - (price * discount/100).

Input Example

List<Double> ordersUSD = Arrays.asList(50.0, 150.0, 300.0); // in USD

Expected Flow

- Function → Convert to INR.
  50 → 4150.0, 150 → 12450.0, 300 → 24900.0.
- Predicate → Detect orders above ₹2000 for free shipping.
- Consumer → Print confirmation messages.
- Supplier → Print coupon codes for each order.
- BiFunction → Apply discount (say 10%) and show final price.

👉 **Learning Outcome:** Demonstrates **currency conversion, free shipping, discounts, and coupons** — all using **wrapper types** + built-in functional interfaces.

✅ These two assignments give **end-to-end practice** of:

- Checking conditions (Predicate)
- Performing actions (Consumer)
- Supplying data (Supplier)
- Transforming data (Function)
- Combining inputs (BiFunction)