Online Payment Processing

An e-commerce platform processes payments based on the following conditions:

1. The user must enter **valid card details**.

2. The user must have **sufficient balance**.

3. The transaction must be **within the allowed transaction limit**.

Suppose you are testing this application using a decision table-based testing technique. Develop the decision table for this and test cases based on this

Conditions								
valid card?	T	T	T	T	F	F	F	F
sufficient balance?	T	T	F	F	T	T	F	F
within the allowed transaction limit?	T	F	T	F	T	F	T	F
Output								
Approved	X							
Declined		X	X	X	X	X	X	X
Impossible								

Test Case #	Inputs	Expected Output	Actual Output
1	Valid card, has sufficient balance, within allowed limit	Approved	
2	Valid card, has sufficient balance, NOT within allowed limit	Declined	
3	Valid card, Does Not have sufficient balance, within allowed limit	Declined	
4	Valid card, Does Not have sufficient balance, NOT within allowed limit	Declined	

5, 6, 7, 8	Invalid Card, …. (if card is invalid then other conditions can't be tested)	Declined	

```
public class PaymentProcessor {
public static String processTransaction(boolean validCard, Boolean
sufficientBalance, boolean withinLimit) {
        if (!validCard) {
            return "Declined: Invalid Card Details";
        }
        if (!sufficientBalance) {
            return "Declined: Insufficient Balance";
        }
```

```java
        if (!withinLimit) {
            return "Declined: Transaction Limit Exceeded";
        }

        return "Approved";
    }
}
public class PaymentProcessorTest {
    @ParameterizedTest
    @CsvSource({
        "true, true, true,  Approved",
        "true, true, false,  Declined: Transaction Limit Exceeded",
         "true, false, true,  Declined: Insufficient Balance",
        "true, false, false,  Declined: Insufficient Balance ",
        "false, true, true,  Declined: Invalid Card Details",
        "false, true, false, Declined: Invalid Card Details",
        "false, false, true, Declined: Insufficient Balance",
        "false, false, false, Declined: Invalid Card Details"
    })
void testProcessTransaction(boolean validCard, boolean sufficientBalance,
boolean withinLimit, String expectedResult)
    {
assertEquals(expectedResult, PaymentProcessor.processTransaction(validCard, sufficientBalance,
withinLimit));
    }

}
```