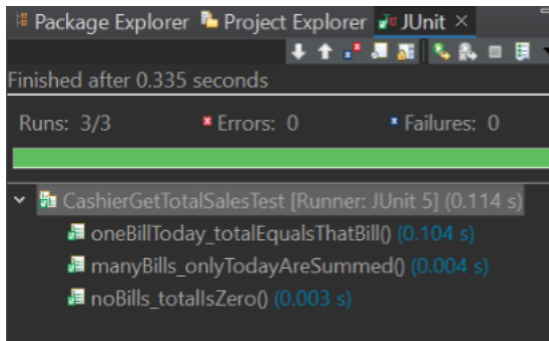


# Testing Analysis

*Antea Koxherri*

## 1. Boundary Value Testing : Method: Cashier.getTotalSales()



**Purpose** This test class applies **Boundary Value Testing (BVT)** to verify the correctness of the `getTotalSales()` method. The focus is on edge cases related to the number of bills and the date-based selection of valid sales.

### Test Case 1: No bills (Minimum boundary)

**Input:** A cashier with no bills recorded

**Expected Result:** The total sales value is 0.0

**Explanation:** This test represents the minimum input boundary (zero bills). The method should correctly return zero when no sales data is available.

### Test Case 2: One bill today (Minimum + 1 boundary)

**Input:** A single bill created on the current day with a total amount of 150.50

**Expected Result:** The total sales equals 150.50

**Explanation:** This test verifies that the method functions correctly when there is exactly one valid bill for today.

### Test Case 3: Multiple bills with mixed dates (Date filtering boundary)

**Input:** Two bills created on the current day with totals 100 and 200

One bill created on the previous day with a total of 999

**Expected Result:** The total sales is 300.0

**Explanation:** This test checks the logical boundary between bills that should be included (today's bills) and bills that should be excluded (bills from previous days).

## Conclusion

These test cases cover the key boundary conditions of the `getTotalSales()` method: no bills present, a single valid bill, and correct exclusion of bills outside the current day. As a result, the tests ensure that the method behaves correctly at its boundary values.

## 2. Code Coverage Testing – Category.checkStockAlert()

**Purpose of testing** :The purpose of Code Coverage Testing is to verify that all decision paths in the `checkStockAlert()` method are executed.

The condition `stockQuantity ≤ minStockLevel` was tested for both outcomes:

TRUE: item included in the alert list , FALSE: item excluded from the alert list.

Asset	Coverage	Instructions	Instructions	Instructions
com.electronic-store	1.0 %	165	16,462	16,627
com.electronic-store.view	0.0 %	78	10,864	16,124
com.electronic-store.view2	0.0 %	0	10,854	10,854
com.electronic-store.view3	0.0 %	0	2,640	2,640
com.electronic-store.view4	0.0 %	0	768	768
com.electronic-store.view5	0.0 %	0	448	448
com.electronic-store.view6	0.0 %	0	325	325
com.electronic-store.view7	0.0 %	0	312	312
com.electronic-store.view8	21.9 %	78	278	356
com.electronic-store.view9	0.0 %	0	115	115
com.electronic-store.view10	22.7 %	30	102	132
com.electronic-store.view11	44.8 %	43	151	194
com.electronic-store.view12	0.0 %	0	221	221
com.electronic-store.view13	17.3 %	87	416	503
com.electronic-store.view14	0.0 %	0	304	304
com.electronic-store.view15	43.7 %	87	112	199
com.electronic-store.view16	0.0 %	0	56	56
com.electronic-store.view17	0.0 %	0	56	56
com.electronic-store.view18	100.0 %	87	0	87

inished after 0.040 seconds

Runs: 2/2      Errors: 0      Failures: 0

CategoryCheckStockAlertCoverageTest [Runner: JUnit 5] (0.098 s)

- coversTrueBranch\_stockLessOrEqualMin\_includedInAlert() (0.093 s)
- coversFalseBranch\_stockGreaterThanMin\_notIncludedInAlert() (0.003 s)

## Purpose of Testing

## Tested Equivalence Classes & Results

**Representative values tested:** two bills created on the current day.

## 2) Old-bills class (billDate < today)

The method returned an empty list, confirming that bills from earlier dates are correctly excluded.

**Representative value tested:** a cashier with no bills recorded.


#### 4) Mixed-bills class (today + old bills)





The method returned only the bill from today and excluded the old bill, confirming correct filtering under mixed input conditions.

Equivalence Class Testing confirmed that `viewDailyBills()` consistently separates today's bills from older bills and safely returns an empty list when no bills are available.

Runs: 4/4      ✖ Errors: 0      ✖ Failures: 0

---

✓  CashierViewDailyBillsTest [Runner: JUnit 5] (0.083 s)

- ✓  noBills\_returnsEmptyList() (0.038 s)
- ✓  hasBillsForToday\_returnsTodayBills() (0.036 s)
- ✓  hasOnlyOldBills\_returnsEmptyList() (0.002 s)
- ✓  mixedBills\_todayAndOld\_returnsOnlyTodayBills() (0.003 s)