Task-1

This operational plan describes the high-efficiency daily micro-audit routine to ensure inventory accuracy and billing discipline at the pharmacy.

1. High-Risk Medicine List

In the time slot of 20 minutes only such "High-Risk" drugs are audited daily.

Medicine (Master)	Why High Risk?
Dolo 650	**High Volume & Name Confusion:** High sales frequency and frequent manual entry variations (e.g., "Dolo kind") increase error risk.
Azithromycin 500	**Confusing Abbreviations:** Frequently typed as "Azithro," which can be confused with different strengths like 250mg.
Pantoprazole 40	**High Frequency:** A top-prescribed item where staff often enter quantities on "autopilot," leading to count errors.

2. Daily Name-Variation Check

We use a Mapping Table to standardize messy HMS sales data before performing the audit.

Entered Name Variant (HMS)	Mapped To Master SKU	Action Required
Dolo / Dolo kind	**Dolo 650**	Add "Dolo kind" to mapping; coach staff to use standard name.
Azithro 500 / Azithromycin	**Azithromycin 500**	Standardize variants; flag missing dosage info as a safety risk.

3. Daily Usage & Variance Audit

We compare the Expected Closing Stock (Opening + Purchases - Mapped Sales) against the Actual Closing Stock recorded at the end of the day.

Audit Calculation (Based on Provided Data):

- **Dolo 650:** 1,000 (Open) + 500 (Buy) - 30 (Sales: 10+15+5) = **1,470 Expected**.
- **Azithromycin 500:** 300 (Open) + 200 (Buy) - 10 (Sales: 6+4) = **490 Expected**.

Sample Audit Table:

Medicine	**Expected**	**Actual**	**Diff**	**Action Rule**
Dolo 650	1,470	1,440	**-30**	**INVESTIGATE:** Large shortage. Check for unbilled bulk sales.
Azithro 500	490	480	**-10**	**REVIEW:** Potential unit error (tablet vs. strip).

4. Clear Action Rules

The above regulations specify the following actions precisely regarding what to do when an error is found.

Check

Rule 1 (Name Error): If a new variant emerges (such as "Dolo kind"), the map list should then be updated, and the billing department should be notified accordingly.

Rule 2 (Stock Variance): If the difference is >2%, investigate the physical bin and check trash for unbilled strips.

Rule 3 (Escalation): Only disrupt the Doctor when there is a high financial consequence, when the problem recurs in spite of retraining the staff member, or in the event of potential health problems for the patient.

5. Operational Checklists

Daily Checklist (20 Minutes)

- **Standardize (7 mins):** Run the script to standardize messy HMS names to match the Master List.
- **Audit (8 mins):** Calculate expected stock and compare it with the actual count.
- **Spot Check (5 mins):** Choose 3 bills randomly to ensure if the quantity match the prescription.
- **Log:** Enter errors into the Daily Tracker.

Weekly Checklist

- **Trend Check:** Check the Error Log. If the same error occurs **≥3 times**, retrain the employees.
- **System Update:** In case various drugs have a similar pattern, it is essential to update the Standard Operating Procedure (SOP)
- **Refresh:** Check the High-Risk List for exchange of drugs if the sales distribution has changed.

6. Automation (Google Apps Script)

This script automates the cleaning and math of the audit.

JavaScript

```
function runInventoryAudit() {
  const ss = SpreadsheetApp.getActiveSpreadsheet();
  const salesSheet = ss.getSheetByName("Sales_Log");
  const mappingSheet = ss.getSheetByName("Mapping_Table");
  const auditSheet = ss.getSheetByName("Inventory_Audit");

  // 1. Load Name Mappings (HMS Variant -> Master Name)
  const mapData = mappingSheet.getDataRange().getValues();
  let nameMap = {};
  for (let i = 1; i < mapData.length; i++) {
    nameMap[mapData[i][0].toLowerCase()] = mapData[i][1];
  }

  // 2. Aggregate Sales from messy HMS data
  const salesData = salesSheet.getDataRange().getValues();
  let salesSums = {};
  for (let i = 1; i < salesData.length; i++) {
    let variant = salesData[i][1].toLowerCase();
    let master = nameMap[variant] || "Unknown";
    salesSums[master] = (salesSums[master] || 0) + salesData[i][2];
  }

  // 3. Update Audit Sheet with Math [cite: 46]
  const auditRows = auditSheet.getDataRange().getValues();
  for (let j = 1; j < auditRows.length; j++) {
    let medName = auditRows[j][0];
    let sold = salesSums[medName] || 0;
    auditSheet.getRange(j + 1, 4).setValue(sold); // Column D: Sales

    let expected = (auditRows[j][1] + auditRows[j][2]) - sold;
    auditSheet.getRange(j + 1, 5).setValue(expected); // Column E: Expected
  }
}
```

I have made the structure and logic. To benefit from this, you can then paste these steps into your document while copying this code into your Google Sheet Apps Script editor.