|  |
| --- |
| --Unreturned Items Report |
| SELECT  ci.Material\_ID,  ci.Challan\_Item\_ID,  r.Project\_Code\_For,  r.Requisition\_Type  FROM Challan\_Items ci  JOIN Requisition r ON ci.Requisition\_ID = r.Requisition\_ID  WHERE ci.Challan\_Item\_ID NOT IN (  SELECT Reference\_Challan\_Item\_ID  FROM Material\_Transaction\_Log  WHERE Transaction\_Type = 'RETURN'  ) |
|  |
| --Stock Movement Report (Material Ledger) |
| SELECT  t.Material\_ID,  t.Serial\_Number,  t.Transaction\_Type,  t.Transaction\_Date,  t.Out\_Quantity,  t.In\_Quantity,  t.Requisition\_ID,  t.Challan\_ID  FROM Material\_Transaction\_Log t  ORDER BY t.Material\_ID, t.Transaction\_Date; |
|  |
| --Material Usage by Requisition Type |
| SELECT  r.Requisition\_Type,  ci.Material\_ID,  SUM(ci.Quantity) AS Total\_Quantity  FROM Requisition r  JOIN Challan\_Items ci ON r.Requisition\_ID = ci.Requisition\_ID  GROUP BY r.Requisition\_Type, ci.Material\_ID  ORDER BY r.Requisition\_Type, ci.Material\_ID; |
|  |
| --Employee-Wise Material Requisition Report |
| SELECT  r.Employee\_ID,  r.Material\_ID,  SUM(ci.Quantity) AS Total\_Quantity\_Requested  FROM Requisition r  JOIN Challan\_Items ci ON r.Requisition\_ID = ci.Requisition\_ID  GROUP BY r.Employee\_ID, r.Material\_ID  ORDER BY r.Employee\_ID, r.Material\_ID; |
|  |
| --Material History Report  -- Full history of a material (replace @MaterialID) |
| DECLARE @MaterialID INT = 8;  SELECT  t.Transaction\_Type,  t.Transaction\_Date,  t.Material\_ID,  t.Serial\_Number,  t.Challan\_ID,  t.Requisition\_ID,  t.Out\_Quantity,  t.In\_Quantity,  t.Remarks  FROM Material\_Transaction\_Log t  WHERE t.Material\_ID = @MaterialID  ORDER BY t.Transaction\_Date; |
|  |
| --Project-Wise Net Consumption Report (After Returns)  --Shows materials consumed per project minus any returned quantities.  -- Project-wise net consumption |
| SELECT  r.Project\_Code\_For,  ci.Material\_ID,  SUM(ci.Quantity) AS Delivered\_Quantity,  ISNULL((  SELECT COUNT(\*)  FROM Material\_Transaction\_Log t  WHERE t.Transaction\_Type = 'RETURN'  AND t.Reference\_Challan\_Item\_ID = ci.Challan\_Item\_ID  ), 0) AS Returned\_Quantity,  SUM(ci.Quantity) - ISNULL((  SELECT COUNT(\*)  FROM Material\_Transaction\_Log t  WHERE t.Transaction\_Type = 'RETURN'  AND t.Reference\_Challan\_Item\_ID = ci.Challan\_Item\_ID  ), 0) AS Net\_Consumption  FROM Challan\_Items ci  JOIN Requisition r ON ci.Requisition\_ID = r.Requisition\_ID  WHERE r.Project\_Code\_For IS NOT NULL  GROUP BY r.Project\_Code\_For, ci.Material\_ID, ci.Challan\_Item\_ID |
|  |
| --Project wise consumption.  -- Replace with your desired Project Code |
| DECLARE @ProjectCode NVARCHAR(50) = '118045';  SELECT  r.Project\_Code\_For,  ci.Material\_ID,  SUM(ci.Quantity) AS Total\_Consumed\_Quantity  FROM Challan\_Items ci  JOIN Requisition r  ON ci.Requisition\_ID = r.Requisition\_ID  WHERE r.Project\_Code\_For = @ProjectCode  GROUP BY  r.Project\_Code\_For,  ci.Material\_ID  ORDER BY  ci.Material\_ID;  --All project at aglance  SELECT  r.Project\_Code\_For,  ci.Material\_ID,  SUM(ci.Quantity) AS Total\_Consumed\_Quantity  FROM Challan\_Item ci  JOIN Requisition r  ON ci.Requisition\_ID = r.Requisition\_ID  WHERE r.Project\_Code\_For IS NOT NULL  GROUP BY  r.Project\_Code\_For,  ci.Material\_ID  ORDER BY  r.Project\_Code\_For,  ci.Material\_ID; |
|  |
| -- Replace this with the Material ID you want to check |
| DECLARE @Material\_ID INT = 8;  SELECT  ci.Material\_ID,  r.Requisition\_ID,  r.Project\_Code\_For,  r.Requisition\_Type  FROM Challan\_Items ci  JOIN Requisition r  ON ci.Requisition\_ID = r.Requisition\_ID  WHERE ci.Material\_ID = @Material\_ID  AND r.Project\_Code\_For IS NOT NULL  GROUP BY  ci.Material\_ID,  r.Requisition\_ID,  r.Project\_Code\_For,  r.Requisition\_Type; |
|  |
| -- Return Status Item List |
| -- 1. Identify Challan Items delivered for 'Test Purpose'  WITH TestPurposeRequisitions AS (  SELECT Requisition\_ID  FROM Requisition  WHERE Requisition\_Type = 'Test Purpose'  ),  DeliveredChallanItems AS (  SELECT ci.Challan\_Item\_ID, ci.Material\_ID  FROM Challan\_Items ci  INNER JOIN TestPurposeRequisitions tpr ON ci.Requisition\_ID = tpr.Requisition\_ID  ),  ReturnedItems AS (  SELECT Reference\_Challan\_Item\_ID  FROM Material\_Transaction\_Log  WHERE Transaction\_Type = 'RETURN' AND Reference\_Challan\_Item\_ID IS NOT NULL  )  -- 2. Final: Get status  SELECT dci.\*,  CASE  WHEN ri.Reference\_Challan\_Item\_ID IS NOT NULL THEN 'Returned'  ELSE 'Not Returned'  END AS Delivery\_Status  FROM DeliveredChallanItems dci  LEFT JOIN ReturnedItems ri ON dci.Challan\_Item\_ID = ri.Reference\_Challan\_Item\_ID; |
|  |
| -- Material wise return list  -- Replace `:material\_id` with the ID you want to check |
| WITH MaterialChallanItems AS (  SELECT Challan\_Item\_ID  FROM Challan\_Items  WHERE Material\_ID = @material\_id  ),  ReturnedItems AS (  SELECT DISTINCT Reference\_Challan\_Item\_ID  FROM Material\_Transaction\_Log  WHERE Transaction\_Type = 'RETURN'  )  SELECT  mci.Challan\_Item\_ID,  CASE  WHEN ri.Reference\_Challan\_Item\_ID IS NOT NULL THEN 'Returned'  ELSE 'Not Returned'  END AS Return\_Status  FROM MaterialChallanItems mci  LEFT JOIN ReturnedItems ri  ON mci.Challan\_Item\_ID = ri.Reference\_Challan\_Item\_ID; |