

Напредни Бази на Податоци

Индекси и оптимизација на прашалници

Проект: WIMS

Андреј Арсовски 231170

Мартина Ивановска 231014

Мила Спасевска 231046

View 1: view_current_warehouse_stock - Shows current stock per warehouse, product, and or variant

1. Примарен филтер за погледот view_current_warehouse_stock ќе биде според warehouse_id, а може да се филтрира и според product_id и variant_id.
2. Ќе биде употребувано од страна на вработените кога сакаат да направат увид во состојбата на одреден warehouse, одреден продукт или конкретна варијанта.
3. Иницијално време за извршување на овој поглед за секој warehouse за секој продукт е доста долго околу 13 секунди поради тоа што е потребен full scan за сите редови во inventory. Но ова не може да се оптимизира поради неопходен seq scan.
4. Иницијално време за извршување на овој поглед по warehouse е : 1.8 секунди. Времето е прифатливо, Иницијално време за извршување на овој поглед по продукт е : 2,8 секунди. Времето е прифатливо, Иницијално време за извршување на овој поглед по варијанта е : 0.3 секунди. Времето е прифатливо.
5. Во продолжение изваддоци од сите планови и времиња на извршување.

```
[2026-05-06 10:00:55] test.public> SELECT *
                                FROM view_current_warehouse_stock
                                WHERE warehouse_id = 14
[2026-05-06 10:00:57] 500 rows retrieved starting from 1 in 1 s 806 ms (execution: 1 s 465 ms, fetching: 341 ms)
```

Operation	Params	Rows	Total Cost	Startup Cost	Raw Desc
Select					JIT = ("Functions"
Aggregate		673237	354850.19	227890.38	Strategy = Sorte
Unknown (Gather Merge)		673236	338019.28	227890.38	Parent Relations
Aggregate		280515	259311.11	226890.36	Strategy = Sorte
Unknown (Incremental Sort)		280515	251596.95	226890.36	Parent Relations
Nested Loops (Nested Loop)		280515	234553.56	226829.63	Parent Relations
Merge Join		280515	231038.96	226829.49	Parent Relations
Sort		280515	227502.48	226801.2	Parent Relations
Hash Join		280515	192786.79	18758.93	Parent Relations
Hash Join		280515	175411.85	6552.36	Parent Relations
Full Scan (Seq Scan)	table: inventory;	4488250	150378.5	0.0	Parent Relations
Transformation (Hash)		23529	6258.25	6258.25	Parent Relations
Nested Loops (Nested Loop)		23529	6258.25	2.7	Parent Relations
Hash Join		4706	2158.58	2.28	Parent Relations
Full Scan (Seq Scan)	table: locations;	75294	1949.94	0.0	Parent Relations
Transformation (Hash)		6	2.2	2.2	Parent Relations
Full Scan (Seq Scan)	table: sections;	6	2.2	0.0	Parent Relations
Index Scan	table: bins; index: uq_bins_location;	5	0.82	0.42	Parent Relations
Transformation (Hash)		167270	8971.7	8971.7	Parent Relations
Full Scan (Seq Scan)	table: product_variants;	167270	8971.7	0.0	Parent Relations
Sort		350	29.16	28.29	Parent Relations
Full Scan (Seq Scan)	table: products;	350	13.5	0.0	Parent Relations
Temporary (Materialize)		1	8.17	0.14	Parent Relations
Index Scan	table: warehouses; index: warehouses_pkey;	1	8.16	0.14	Parent Relations

```
[2026-05-10 19:06:36] test.public> SELECT *
                                FROM view_current_warehouse_stock
                                WHERE product_id = 2
[2026-05-10 19:06:39] 500 rows retrieved starting from 1 in 2 s 938 ms (execution: 2 s 622 ms, fetching: 316 ms)
```

Operation	Params	Rows	Total Cost	Startup Cost	Raw Desc
Select					JIT = ("Functions":60,"Options":
Nested Loops (Nested Loop)		617485	489034.77	284931.95	Parallel Aware = false;Async Ca
Index Scan	table: products; index: products_pkey;	1	8.17	0.15	Parent Relationship = OuterPara
Hash Join		617485	482851.76	284931.8	Parent Relationship = InnerPara
Hash Join		771856	480769.11	284918.2	Parent Relationship = OuterPara
Aggregate		771856	444137.38	266233.62	Strategy = Sorted-Partial Mode
Unknown (Gather Merge)		1294514	417001.11	266233.62	Parent Relationship = OuterPara
Sort		539381	266582.05	265233.6	Parent Relationship = OuterPara
Hash Join		539381	200973.98	25820.41	Parent Relationship = OuterPara
Hash Join		539381	199492.43	25817.25	Parent Relationship = OuterPara
Hash Join		539381	195185.4	22926.13	Parent Relationship = OuterPara
Hash Join		539381	171805.51	9641.12	Parent Relationship = OuterPara
Full Scan (Seq Scan)	table: inventory;	4488575	150381.75	0.0	Parent Relationship = OuterPara
Transformation (Hash)		20100	9389.88	9389.88	Parent Relationship = InnerPara
Full Scan (Seq Scan)	table: product_variants;	20100	9389.88	0.0	Parent Relationship = OuterPara
Transformation (Hash)		266667	8648.67	8648.67	Parent Relationship = InnerPara
Full Scan (Seq Scan)	table: bins;	266667	8648.67	0.0	Parent Relationship = OuterPara
Transformation (Hash)		75294	1949.94	1949.94	Parent Relationship = InnerPara
Full Scan (Seq Scan)	table: locations;	75294	1949.94	0.0	Parent Relationship = OuterPara
Transformation (Hash)		96	1.96	1.96	Parent Relationship = InnerPara
Full Scan (Seq Scan)	table: sections;	96	1.96	0.0	Parent Relationship = OuterPara
Transformation (Hash)		401448	11313.48	11313.48	Parent Relationship = InnerPara
Full Scan (Seq Scan)	table: product_variants;	401448	11313.48	0.0	Parent Relationship = OuterPara
Transformation (Hash)		160	11.6	11.6	Parent Relationship = InnerPara
Full Scan (Seq Scan)	table: warehouses;	160	11.6	0.0	Parent Relationship = OuterPara

```
[2026-05-10 19:07:26] test.public> SELECT *
                                FROM view_current_warehouse_stock
                                WHERE variant_id = 10000
[2026-05-10 19:07:27] 13 rows retrieved starting from 1 in 360 ms (execution: 8 ms, fetching: 352 ms)
```

Operation	Params	Rows	Total Cost	Startup Cost	Raw Desc
Select					
Nested Loops (Nested Loop)		16	387.06	363.91	Parallel Aware = false; Async C...
Index Scan	table: product_variants; index: product_variants_pkey;	1	8.44	0.42	Parent Relationship = Outer; Para...
Hash Join		16	378.46	363.49	Parent Relationship = Inner; Para...
Full Scan (Seq Scan)	table: products;	350	13.5	0.0	Parent Relationship = Outer; Para...
Transformation (Hash)		16	363.29	363.29	Parent Relationship = Inner; Para...
Hash Join		16	363.29	350.93	Parent Relationship = Outer; Para...
Full Scan (Seq Scan)	table: warehouses;	180	11.6	0.0	Parent Relationship = Outer; Para...
Transformation (Hash)		16	350.73	350.73	Parent Relationship = Inner; Para...
Aggregate		16	350.73	350.18	Strategy = Sorted; Partial Mode ...
Sort		26	350.24	350.18	Parent Relationship = Outer; Para...
Nested Loops (Nested Loop)		26	340.56	1.72	Parent Relationship = Outer; Para...
Index Scan	table: product_variants; index: product_variants_pkey;	1	8.44	0.42	Parent Relationship = Outer; Para...
Nested Loops (Nested Loop)		26	340.86	1.29	Parent Relationship = Inner; Para...
Nested Loops (Nested Loop)		26	336.7	1.15	Parent Relationship = Outer; Para...
Nested Loops (Nested Loop)		26	328.39	0.86	Parent Relationship = Outer; Para...
Index Scan	table: inventory; index: uq_inventory_variant_bin;	26	108.88	0.43	Parent Relationship = Outer; Para...
Index Scan	table: bins; index: bins_pkey;	1	8.44	0.42	Parent Relationship = Inner; Para...
Index Scan	table: locations; index: locations_pkey;	1	0.32	0.29	Parent Relationship = Inner; Para...
Index Scan	table: sections; index: sections_pkey;	1	0.16	0.14	Parent Relationship = Inner; Para...

View 2 : view_products_by_category - Lists all products grouped by their categories

1. Примарен филтер за погледот view_products_by_category ќе биде според main_category_id, а може да се филтрира и според product_id доколку сакаме да видиме даден продукт во кој се категории се наоѓа.
2. Ќе биде употребувано од страна на вработените или клиенти кога сакаат да разгледуваат продукти кои се дел од одбрана категорија. Корисно ќе биде и понатаму кога ќе сакаме да добиеме детални информации за одредена варијанта.
3. Иницијално време за извршување на овој поглед за секој warehouse за секој продукт и категорија е доста кратко околу 350 ms.
4. Иницијално време за извршување на овој поглед по category_id е : 1.8 секунди. Времето е прифатливо, Иницијално време за извршување на овој поглед по product_id е : 350 ms. Времето е прифатливо
5. Во продолжение извадоци од сите планови и времиња на извршување.

```
[2026-05-10 19:50:29] test.public> SELECT *
FROM view_products_by_category
[2026-05-10 19:50:29] 75 rows retrieved starting from 1 in 334 ms (execution: 4 ms, fetching: 330 ms)

[2026-05-10 19:51:18] test.public> SELECT *
FROM view_products_by_category
WHERE product_id = 5
[2026-05-10 19:51:18] 5 rows retrieved starting from 1 in 358 ms (execution: 7 ms, fetching: 351 ms)
```

```
[2026-05-10 19:46:32] test.public> SELECT *
      FROM view_products_by_category
      WHERE main_category_id = 5
[2026-05-10 19:46:32] 6 rows retrieved starting from 1 in 327 ms (execution: 3 ms, fetching: 324 ms)
```

Operation	Params	Rows	Total Cost	Startup Cost	Raw Desc
Select					
Hash Join		4292	7568.13	5442.24	Parallel Aware = false; Async Cap...
Nested Loops (Nested Loop)		464	7445.7	5343.25	Parent Relationship = Outer; Parall...
Index Scan	table: categories; index: categories_pkey;	1	8.17	0.15	Parent Relationship = Outer; Parall...
Access (CTE Scan)	table: descendants;	464	7432.9	5343.1	Parent Relationship = Inner; Parall...
Union (Recursive Union)		92880	5343.1	0.0	Parent Relationship = InitPlan; Sub...
Full Scan (Seq Scan)	table: categories;	430	14.3	0.0	Parent Relationship = Outer; Parall...
Hash Join		9245	440.0	19.68	Parent Relationship = Inner; Parall...
Access (WorkTable Scan)	table: descendants;	4300	86.0	0.0	Parent Relationship = Outer; Parall...
Transformation (Hash)		430	14.3	14.3	Parent Relationship = Inner; Parall...
Full Scan (Seq Scan)	table: categories;	430	14.3	0.0	Parent Relationship = Outer; Parall...
Transformation (Hash)		1850	75.86	75.86	Parent Relationship = Inner; Parall...
Hash Join		1850	75.86	37.55	Parent Relationship = Outer; Parall...
Hash Join		1850	53.08	19.68	Parent Relationship = Outer; Parall...
Full Scan (Seq Scan)	table: product_categories;	1850	28.5	0.0	Parent Relationship = Outer; Parall...
Transformation (Hash)		430	14.3	14.3	Parent Relationship = Inner; Parall...
Full Scan (Seq Scan)	table: categories;	430	14.3	0.0	Parent Relationship = Outer; Parall...
Transformation (Hash)		350	13.5	13.5	Parent Relationship = Inner; Parall...
Full Scan (Seq Scan)	table: products;	350	13.5	0.0	Parent Relationship = Outer; Parall...

View 3: view_variants_per_product - Shows all variants for each product with detailed variant information

1. Примарен филтер за погледот view_variants_per_product ќе биде според product_id
2. Ќе биде употребувано од страна на вработените или клиенти кога сакаат да разгледуваат конкретни варијанти од одреден тип на производи. Овој поглед заедно со поглед број 2 би бил употребен во еден типичен flow на самата апликација. Пример: корисникот бира категорија -> бира продукт -> листа од сите варијанти за тој продукт.
3. Иницијално време за извршување на овој поглед за секоја варијанта на еден продукт е доста кратко околу 400 ms.
4. Иницијално време за извршување на овој поглед по product_id е : 430 ms. Времето е прифатливо
5. Во продолжение извадоци од сите планови и времиња на извршување.

```
[2026-05-10 20:35:12] test.public> SELECT *
      FROM view_variants_per_product
      WHERE product_id = 13
[2026-05-10 20:35:12] 500 rows retrieved starting from 1 in 434 ms (execution: 22 ms, fetching: 412 ms)
```

Operation	Params	Rows	Total Cost	Startup Cost
Select				
Unknown (Gather Merge)		1499	10609.49	10434.9
Sort		625	9436.44	9434.88
Nested Loops (Nested Loop)		625	9405.85	0.15
Full Scan (Seq Scan)	table: product_variants;	625	9389.88	0.0
Temporary (Materialize)		1	8.17	0.15
Index Scan	table: products; index: products_pkey;	1	8.17	0.15

View 4: view_inventory_summary – View inventory stock for a given product_variant

1. Примарен филтер за погледот view_products_by_category ќе биде според product_variant_id, available quantity.
2. Ќе биде употребувано од страна на вработените и менаџерите кога сакаат да проверат во колкава количина имаме одредена варијанта.
3. Иницијално време за извршување на овој поглед за секој inventory за секој продукт и категорија е околу 3 s што не е лошо за аналитички тип на query.
4. Иницијално време за извршување на овој поглед по product_variant_id е : 320 ms. Времето е прифатливо.
Иницијално време за извршување на овој поглед по available_quantity е : 3,7 s. Времето не е прифатливо.
5. Во продолжение извадоци од сите планови и времиња на извршување.

```
[2026-05-06 11:25:55] 1 row retrieved starting from 1 in 321 ms (execution: 6 ms, fetching: 315 ms)
[2026-05-06 11:26:02] test.public> SELECT *
                                FROM view_inventory_summary where product_variant_id=312312
[2026-05-06 11:26:03] 1 row retrieved starting from 1 in 316 ms (execution: 2 ms, fetching: 314 ms)
```

Operation	Params	Rows	Total Cost	Startup Cost
Select				
Aggregate		1	108.01	4.64
Bitmap Index Scan (Bitmap Heap Scan)	table: inventory;	26	107.74	4.64
Bitmap Index Scan	index: uq_inventory_variant_bin;	26	4.63	0.0

```
[2026-05-10 20:44:07] completed in 0 ms
[2026-05-10 20:46:19] test.public> SELECT *
                                FROM view_inventory_summary
                                WHERE available_quantity < 10
[2026-05-10 20:46:22] 24 rows retrieved starting from 1 in 3 s 701 ms (execution: 3 s 385 ms, fetching: 316 ms)
[2026-05-10 20:48:50] test.public> SELECT *
```

Operation	Params	Rows	Total Cost	Startup Cost	Raw Desc
↳ Select					JIT = ("Functions":8,"Options":{"I...
↳ Aggregate		139019	701308.83	569510.37	Strategy = Sorted;Partial Mode = ...
↳ Unknown (Gather Merge)		1000939	686086.22	569510.37	Parent Relationship = Outer;Parall...
↳ Sort		417058	569553.0	568510.35	Parent Relationship = Outer;Parall...
↳ Aggregate		417058	519599.72	471595.4	Strategy = Hashed;Partial Mode ...
↳ Full Scan (Seq Scan)	table: inventory;	4488575	150381.75	0.0	Parent Relationship = Outer;Parall...

View 5: view_employee_current_warehouse - Detailed warehouse where an employee is assigned or employees per warehouse

1. Примарен филтер за погледот view_products_by_category ќе биде според employee_id, а може да се филтрира и според warehouse_id доколку сакаме да видиме кои се вработени во одреден warehouse, и исто така може да се филтрира додатно по employment_status за да се земат активни вработени.
2. Ќе биде употребувано од страна на администратори кога сакаат да видат кои се задачите што ги има еден вработен моментално, кој е примарниот warehouse на одреден вработен или пак да ги види сите вработени кои работат во даден warehouse.
3. Иницијално време за извршување на овој поглед за секој warehouse за секој продукт и категорија е доста кратко околу 400 ms.
4. Иницијално време за извршување на овој поглед по employee_id е : 300 ms
Времето е прифатливо
Иницијално време за извршување на овој поглед по warehouse_id е : 350 ms.
Времето е прифатливо
5. Во продолжение изваддоци од сите планови и времиња на извршување.

```
[2026-05-06 11:13:24] test.public> SELECT * FROM view_employee_current_warehouse where warehouse_id=13
[2026-05-06 11:13:25] 114 rows retrieved starting from 1 in 357 ms (execution: 7 ms, fetching: 350 ms)
```

Operation	Params	Rows	Total Cost	Startup Cost
↳ Select				
↳ Nested Loops (Nested Loop)		42	542.92	341.55
↳ Nested Loops (Nested Loop)		42	516.88	341.27
↳ Index Scan	table: warehouses; index: warehouses_pkey;	1	8.16	0.14
↳ Hash Join		42	508.29	341.12
↳ Full Scan (Seq Scan)	table: employees;	1800	151.0	0.0
↳ Transformation (Hash)		47	340.54	340.54
↳ Full Scan (Seq Scan)	table: employee_warehouse_assignments;	47	340.54	0.0
↳ Index Scan	table: employees; index: employees_pkey;	1	0.61	0.28

```
[2026-05-06 11:14:45] test.public> SELECT * FROM view_employee_current_warehouse where employee_id=13
[2026-05-06 11:14:45] 1 row retrieved starting from 1 in 331 ms (execution: 1 ms, fetching: 330 ms)
```

Session: test.public.view_emp...yee_current_warehouse

Operation	Params	Rows	Total Cost	Startup Cost
Select		1	45.67	0.98
Nested Loops (Nested Loop)		1	37.36	0.71
Nested Loops (Nested Loop)		1	29.1	0.56
Nested Loops (Nested Loop)		1	8.29	0.28
Index Scan	table: employees; index: employees_pkey;	1	20.79	0.28
Index Scan	table: employee_warehouse_assignments; index: uq_employee_assignments_unique_period;	1	8.16	0.14
Index Scan	table: warehouses; index: warehouses_pkey;	1	8.29	0.28
Index Scan	table: employees; index: employees_pkey;	1		

View 6: view_employee_permissions - Lists employee roles and their associated permissions

1. Примарен филтер за погледот view_employee_permissions ќе биде според employee_id.
2. Ќе биде употребувано од страна на администратори кога сакаат да видат кој пермисии ги има даден вработен и дали му е дозволено да извршува одредени задачи.
3. Иницијално време за извршување на овој поглед за секој вработен за секој пермисија е доста кратко околу 350 ms.
4. Иницијално време за извршување на овој поглед по employee_id е : 300 ms. Времето е прифатливо.
5. Во продолжение изваддоци од сите планови и времиња на извршување.

```
[2026-05-06 11:09:52] test.public> SELECT *
FROM view_employee_permissions
WHERE employee_id = 22
[2026-05-06 11:09:52] 10 rows retrieved starting from 1 in 341 ms (execution: 7 ms, fetching: 334 ms)
```

Output: Plan

Operation	Params	Rows	Total Cost	Startup Cost
Select		1	28.5	0.87
Nested Loops (Nested Loop)		1	28.02	0.72
Nested Loops (Nested Loop)		1	19.71	0.44
Nested Loops (Nested Loop)		11	14.28	0.29
Full Scan (Seq Scan)	table: permissions_roles;	153	2.53	0.0
Unknown (Memoize)		1	0.62	0.29
Index Scan (Index Only)	table: roles_employees; index: roles_employees_pkey;	1	0.61	0.28
Index Scan	table: roles; index: roles_pkey;	1	0.48	0.15
Index Scan	table: employees; index: employees_pkey;	1	8.29	0.28
Index Scan	table: permissions; index: permissions_pkey;	1	0.48	0.15

View 7:view_inventory_movements_detailed - Detailed inventory movement records including bins and responsible employee

1. Примарен филтер за погледот view_inventory_movements_detailed ќе биде според employee_id, но може и според transaction_type, според created_at, inventory_transaction_id.
2. Ќе биде употребувано од страна на администратори и warehouse менаџери кога сакаат да следат движење на залихи и да видат кој вработен ја извршил секоја трансакција и секое движење.
3. Иницијално време за извршување на овој поглед без филтрирање е повисоко 3.2 s Што е нормално за поголема табела.
4. Иницијално време за извршување на овој поглед по employee_id е за околу 900 ms. Што е прифатливо. Иако после индексот додаден поради другите табели се подобри и стана околу 400 ms. Иницијално време за извршување на овој поглед по transaction_type е за околу 2.7 s. Што е прифатливо. Сепак и со индекси не можеше да се подобри. Иницијално време за извршување на овој поглед по transaction_type е за околу 3.3 s. После индексот додаден поради другите табели се подобри и стана околу 1.7 s. Што е двојно побрзо. Иницијално време за извршување на овој поглед по inventory_transaction_id е за околу 1 s. После индексот додаден поради другите табели се подобри и стана околу 300 ms Што е големо подобрување.
5. Во продолжение извадоци од сите планови и времиња на извршување.

```
CREATE INDEX idx_im_inventory_transactions_id
  ON inventory_movements (inventory_transactions_id);
drop index idx_im_inventory_transactions_id;

CREATE INDEX idx_im_created_at
  ON inventory_movements (created_at);
drop index idx_im_created_at;
```

ПРЕД ИНДЕКС

```

[2026-05-10 21:38:41] completed in 7 ms
[2026-05-10 21:38:52] test.public> SELECT *
                                FROM view_inventory_movements_detailed
                                WHERE employee_id = 431
[2026-05-10 21:38:53] 500 rows retrieved starting from 1 in 891 ms (execution: 552 ms, fetching: 339 ms)

```

Operation	Params	Rows	Total Cost	Startup Cost
Select				
Unknown (Gather)		10013	557058.44	281982.84
Nested Loops (Nested Loop)		4172	555057.14	280982.84
Nested Loops (Nested Loop)		4172	553183.04	280982.42
Nested Loops (Nested Loop)		4172	551329.8	280981.99
Hash Join		4172	551269.35	280981.72
Full Scan (Seq Scan)	table: inventory_movements;	7547139	250476.39	0.0
Transformation (Hash)		4169	280929.6	280929.6
Full Scan (Seq Scan)	table: inventory_transactions;	4169	280929.6	0.0
Temporary (Materialize)		1	8.3	0.28
Index Scan	table: employees; index: employees_pkey;	1	8.29	0.28
Index Scan	table: bins; index: bins_pkey;	1	0.44	0.42
Index Scan	table: bins; index: bins_pkey;	1	0.44	0.42

```

[2026-05-10 21:48:12] test.public> SELECT *
                                FROM view_inventory_movements_detailed
                                WHERE transaction_type = 'TRANSFER'
[2026-05-10 21:48:15] 500 rows retrieved starting from 1 in 2 s 704 ms (execution: 2 s 357 ms, fetching: 347 ms)

```

Operation	Params	Rows	Total Cost	Startup Cost
Select				
Unknown (Gather)		6047451	1547373.95	356021.38
Hash Join		2519771	941628.85	355021.38
Hash Join		2519771	922225.58	354845.38
Hash Join		2519771	836003.15	341040.37
Hash Join		2519771	759624.72	327235.37
Full Scan (Seq Scan)	table: inventory_movements;	7544978	250454.78	0.0
Transformation (Hash)		2519771	280973.23	280973.23
Full Scan (Seq Scan)	table: inventory_transactions;	2519771	280973.23	0.0
Transformation (Hash)		266667	8648.67	8648.67
Full Scan (Seq Scan)	table: bins;	266667	8648.67	0.0
Transformation (Hash)		266667	8648.67	8648.67
Full Scan (Seq Scan)	table: bins;	266667	8648.67	0.0
Transformation (Hash)		2000	151.0	151.0
Full Scan (Seq Scan)	table: employees;	2000	151.0	0.0

```

[2026-05-10 21:47:40] 500 rows retrieved starting from 1 in 1 s 877 ms (execution: 1 s 26 ms, fetching: 864 ms)
[2026-05-10 21:49:52] test.public> SELECT *
                                FROM view_inventory_movements_detailed
                                WHERE created_at >= '2025-01-01 00:00:00.000000'
[2026-05-10 21:49:55] 500 rows retrieved starting from 1 in 3 s 384 ms (execution: 3 s 38 ms, fetching: 346 ms)

```

Operation	Params	Rows	Total Cost	Startup Cost
Select		3014683	1102570.81	326071.71
Unknown (Gather)		3014683	1102570.81	326071.71
Hash Join		1256118	800102.51	325071.71
Hash Join		1256118	790341.63	324895.71
Hash Join		1256118	739522.3	311090.71
Hash Join		1256118	693608.97	297285.7
Full Scan (Seq Scan)	table: inventory_transactions;	7544978	262110.78	0.0
Transformation (Hash)		1256118	269317.23	269317.23
Full Scan (Seq Scan)	table: inventory_movements;	1256118	269317.23	0.0
Transformation (Hash)		266667	8648.67	8648.67
Full Scan (Seq Scan)	table: bins;	266667	8648.67	0.0
Transformation (Hash)		266667	8648.67	8648.67
Full Scan (Seq Scan)	table: bins;	266667	8648.67	0.0
Transformation (Hash)		2000	151.0	151.0
Full Scan (Seq Scan)	table: employees;	2000	151.0	0.0

```
[2026-05-11 09:31:15] test.public> SELECT *
                                FROM view_inventory_movements_detailed
                                WHERE inventory_transaction_id = 3200376
[2026-05-11 09:31:16] 1 row retrieved starting from 1 in 894 ms (execution: 576 ms, fetching: 318 ms)
```

Operation	Params	Rows	Total Cost	Startup Cost
Select		1	270346.47	1001.56
Nested Loops (Nested Loop)		1	270338.16	1001.29
Nested Loops (Nested Loop)		1	270329.69	1000.85
Nested Loops (Nested Loop)		1	270321.25	1000.42
Unknown (Gather)		1	270312.81	1000.0
Full Scan (Seq Scan)	table: inventory_movements;	1	269312.71	0.0
Index Scan	table: bins; index: bins_pkey;	1	8.44	0.42
Index Scan	table: bins; index: bins_pkey;	1	8.44	0.42
Index Scan	table: inventory_transactions; index: inventory_transactions_pkey;	1	8.46	0.44
Index Scan	table: employees; index: employees_pkey;	1	8.29	0.28

ПОСЛЕ ИНДЕКС

```
[2026-05-10 21:41:18] test.public> SELECT *
                                FROM view_inventory_movements_detailed
                                WHERE employee_id = 431
[2026-05-10 21:41:19] 500 rows retrieved starting from 1 in 405 ms (execution: 55 ms, fetching: 350 ms)
```

Operation	Params	Rows	Total Cost	Startup Cost	Notes
Select					JIT = ("Functions":21,"Opt...
Unknown (Gather)		10011	320078.85	1001.57	Parallel Aware = false;As...
Nested Loops (Nested Loop)		4171	318077.75	1.57	Parent Relationship = Out...
Nested Loops (Nested Loop)		4171	316204.1	1.14	Parent Relationship = Out...
Nested Loops (Nested Loop)		4171	314351.3	0.72	Parent Relationship = Out...
Nested Loops (Nested)		4171	314290.86	0.44	Parent Relationship = Out...
Full Scan (Seq Scan)	table: inventory_transactions;	4171	280973.23	0.0	Parent Relationship = Out...
Index Scan	table: inventory_movements; index: idx_inv_move_inv_trans;	1	7.98	0.44	Parent Relationship = Inn...
Temporary (Materialize)		1	8.3	0.28	Parent Relationship = Inn...
Index Scan	table: employees; index: employees_pkey;	1	8.29	0.28	Parent Relationship = Out...
Index Scan	table: bins; index: bins_pkey;	1	0.44	0.42	Parent Relationship = Inn...
Index Scan	table: bins; index: bins_pkey;	1	0.44	0.42	Parent Relationship = Inn...

```
[2026-05-10 22:21:00] test.public> SELECT *
      FROM view_inventory_movements_detailed
     WHERE created_at >= '2025-01-01 00:00:00.000000'
[2026-05-10 22:21:02] 500 rows retrieved starting from 1 in 1 s 745 ms (execution: 1 s 400 ms, fetching: 345 ms)
```

Operation	Params	Rows	Total Cost	Startup Cost
Select				
Unknown (Gather)		2889624	908658.36	148443.33
Hash Join		1204010	618695.96	147443.33
Hash Join		1204010	609332.68	147267.33
Hash Join		1204010	559972.14	133462.33
Hash Join		1204010	515315.6	119657.32
Full Scan (Seq Scan)	table: inventory_transactions;	7547387	262134.87	0.0
Transformation (Hash)		1204010	92849.2	92849.2
Index Scan	table: inventory_movements; index: idx_im_created_at;	1204010	92849.2	0.44
Transformation (Hash)		266667	8648.67	8648.67
Full Scan (Seq Scan)	table: bins;	266667	8648.67	0.0
Transformation (Hash)		266667	8648.67	8648.67
Full Scan (Seq Scan)	table: bins;	266667	8648.67	0.0
Transformation (Hash)		2000	151.0	151.0
Full Scan (Seq Scan)	table: employees;	2000	151.0	0.0

```
[2026-05-11 09:32:33] test.public> SELECT *
      FROM view_inventory_movements_detailed
     WHERE inventory_transaction_id = 3200376
[2026-05-11 09:32:33] 1 row retrieved starting from 1 in 342 ms (execution: 9 ms, fetching: 333 ms)
```

Operation	Params	Rows	Total Cost	Startup Cost
Select				
Nested Loops (Nested Loop)		1	42.11	2.0
Nested Loops (Nested Loop)		1	33.81	1.73
Nested Loops (Nested Loop)		1	25.34	1.29
Nested Loops (Nested)		1	16.9	0.86
Index Scan	table: inventory_movements; index: idx_im_inventory_transactions_id;	1	8.46	0.44
Index Scan	table: bins; index: bins_pkey;	1	8.44	0.42
Index Scan	table: bins; index: bins_pkey;	1	8.44	0.42
Index Scan	table: inventory_transactions; index: inventory_transactions_pkey;	1	8.46	0.44
Index Scan	table: employees; index: employees_pkey;	1	8.29	0.28

VIEW 8:view_variant_details - Detailed product variant view

1. Примарен филтер за погледот view_variant_details ќе биде според variant_id.
2. Ќе биде употребувано од страна на вработените или клиенти кога сакаат да разгледуваат одредена продукт варијанта и сите детални податоци.
3. Иницијално време за извршување на овој поглед за секоја варијанта со неговите детали е доста кратко околу 1 s.
4. Иницијално време за извршување на овој поглед по variant_id е : 350ms. Времето е прифатливо.
5. Во продолжение извадоци од сите планови и времиња на извршување.

```
[2026-05-11 10:27:26] test.public> SELECT *  
      FROM view_variant_details  
     where variant_id = 291588  
[2026-05-11 10:27:27] 15 rows retrieved starting from 1 in 363 ms (execution: 9 ms, fetching: 354 ms)
```

Operation	Params	Rows	Total Cost	Startup Cost
Select				
Nested Loops (Nested Loop)		53	60.0	6.18
Nested Loops (Nested Loop)		53	49.88	6.03
Nested Loops (Nested Loop)		11	37.63	1.29
Nested Loops (Nested Loop)		5	25.41	0.87
Nested Loops (Nested Loop)		1	24.85	0.72
Nested Loops (Nested Loop)		1	16.65	0.57
Index Scan	table: product_variants; index: product_variants_pkey;	1	8.44	0.42
Index Scan	table: products; index: products_pkey;	1	8.17	0.15
Index Scan	table: brands; index: brands_pkey;	1	8.17	0.15
Index Scan (Index Only Scan)	table: product_categories; index: product_categories_pkey;	9	0.45	0.15
Temporary (Materialize)		2	12.1	0.42
Index Scan	table: product_images; index: uq_product_images_variant_position;	2	12.09	0.42
Temporary (Materialize)		5	11.58	4.74
Nested Loops (Nested Loop)		5	11.55	4.74
Hash Join		5	6.45	4.59
Full Scan (Seq Scan)	table: attribute_values;	56	1.56	0.0
Transformation (Hash)		5	4.52	4.52
Index Scan (Index Only Scan)	table: variant_attributes; index: variant_attributes_pkey;	5	4.52	0.43
Index Scan	table: attributes; index: attributes_pkey;	1	1.02	0.14
Index Scan	table: categories; index: categories_pkey;	1	0.19	0.15

VIEW 9: view_category_tree - Category hierarchy:

1. Примарен филтер за погледот view_category_tree или нема да имаме или ќе филтрираме според category_id доколку сакаме да видиме детална патека за дадена категорија.

2. Ќе биде употребувано од страна на вработените или клиенти кога сакаат да ја разгледуваат хиерархијата на категориите.
3. Иницијално време за извршување на овој поглед за секој warehouse за секој продукт и категорија е доста кратко околу 330 ms.
4. Иницијално време за извршување на овој поглед по category_id е : 350 ms. Времето е прифатливо.
5. Во продолжение извадоци од сите планови и времиња на извршување.

```
[2026-05-11 10:32:05] 15 rows retrieved starting from 1 in 378 ms (execution: 7 ms, fetching: 371 ms)
[2026-05-11 10:33:42] test.public> SELECT *
                                FROM view_category_tree
[2026-05-11 10:33:43] 15 rows retrieved starting from 1 in 330 ms (execution: 6 ms, fetching: 324 ms)
[2026-05-11 10:34:22] test.public> SELECT *
```

Operation	Params	Rows	Total Cost	Startup Cost
↳ Select				
↳ Access (CTE Scan)	table: tree;	432	195.61	186.97
↳ Union (Recursive Union)		432	186.97	4.16
↳ Bitmap Index Scan (Bitmap Heap Scan)	table: categories;	2	9.5	4.16
↳ Bitmap Index Scan	index: uq_categories_parent_name;	2	4.16	0.0
↳ Hash Join		43	17.32	0.65
↳ Full Scan (Seq Scan)	table: categories;	430	14.3	0.0
↳ Transformation (Hash)		20	0.4	0.4
↳ Access (WorkTable Scan)	table: tree;	20	0.4	0.0

```
[2026-05-11 10:35:46] test.public> SELECT *
                                FROM view_category_tree
                                WHERE category_id=8
[2026-05-11 10:35:46] 1 row retrieved starting from 1 in 340 ms (execution: 4 ms, fetching: 336 ms)
```

Operation	Params	Rows	Total Cost	Startup Cost
↳ Select				
↳ Access (CTE Scan)	table: tree;	2	196.69	186.97
↳ Union (Recursive Union)		432	186.97	4.16
↳ Bitmap Index Scan (Bitmap Heap Scan)	table: categories;	2	9.5	4.16
↳ Bitmap Index Scan	index: uq_categories_parent_name;	2	4.16	0.0
↳ Hash Join		43	17.32	0.65
↳ Full Scan (Seq Scan)	table: categories;	430	14.3	0.0
↳ Transformation (Hash)		20	0.4	0.4
↳ Access (WorkTable Scan)	table: tree;	20	0.4	0.0