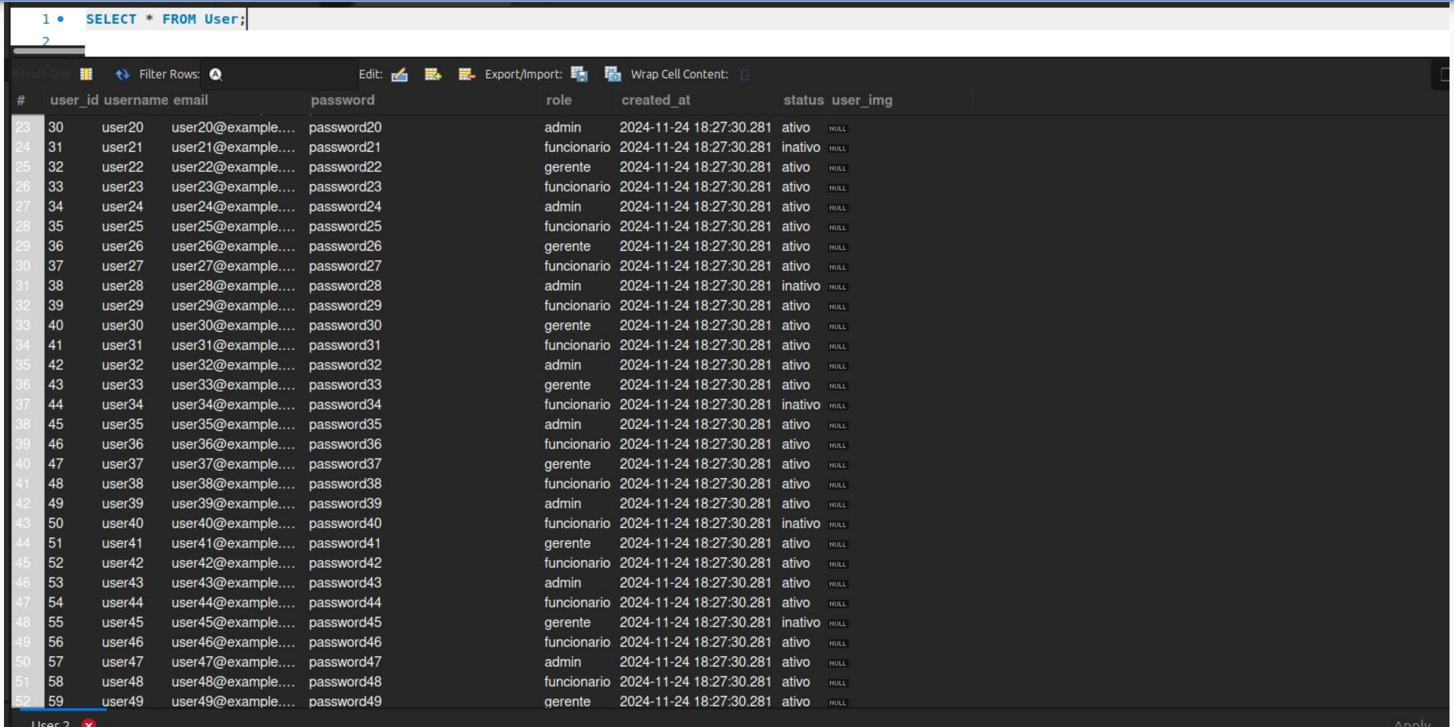
2º Prova de BDR - FATEC 2º DSM

João Gabriel Barros Guimarães - 2º DSM

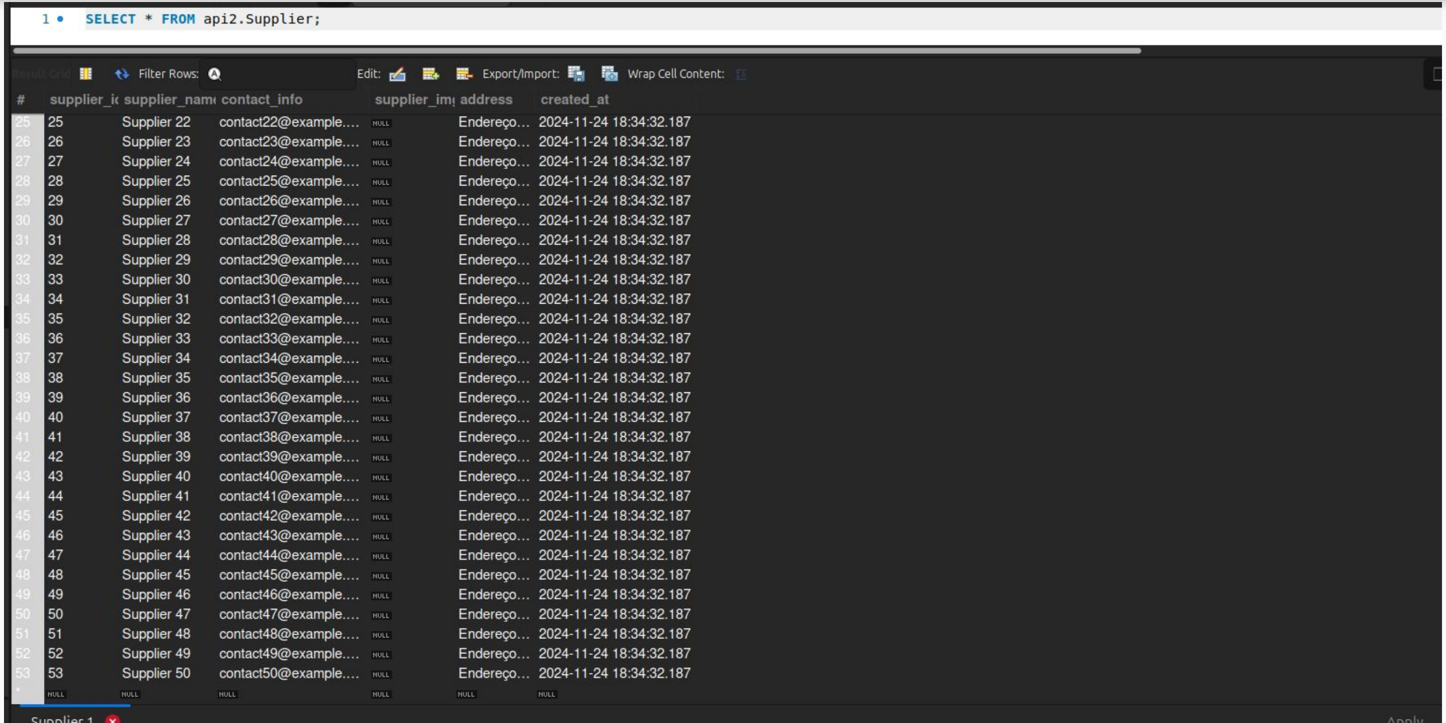
DML - INSERTS tabela User



```sql
SELECT * FROM User;
```

#	user_id	username	email	password	role	created_at	status	user_img
23	30	user20	user20@example....	password20	admin	2024-11-24 18:27:30.281	ativo	NULL
24	31	user21	user21@example....	password21	funcionario	2024-11-24 18:27:30.281	inativo	NULL
25	32	user22	user22@example....	password22	gerente	2024-11-24 18:27:30.281	ativo	NULL
26	33	user23	user23@example....	password23	funcionario	2024-11-24 18:27:30.281	ativo	NULL
27	34	user24	user24@example....	password24	admin	2024-11-24 18:27:30.281	ativo	NULL
28	35	user25	user25@example....	password25	funcionario	2024-11-24 18:27:30.281	ativo	NULL
29	36	user26	user26@example....	password26	gerente	2024-11-24 18:27:30.281	ativo	NULL
30	37	user27	user27@example....	password27	funcionario	2024-11-24 18:27:30.281	ativo	NULL
31	38	user28	user28@example....	password28	admin	2024-11-24 18:27:30.281	inativo	NULL
32	39	user29	user29@example....	password29	funcionario	2024-11-24 18:27:30.281	ativo	NULL
33	40	user30	user30@example....	password30	gerente	2024-11-24 18:27:30.281	ativo	NULL
34	41	user31	user31@example....	password31	funcionario	2024-11-24 18:27:30.281	ativo	NULL
35	42	user32	user32@example....	password32	admin	2024-11-24 18:27:30.281	ativo	NULL
36	43	user33	user33@example....	password33	gerente	2024-11-24 18:27:30.281	ativo	NULL
37	44	user34	user34@example....	password34	funcionario	2024-11-24 18:27:30.281	inativo	NULL
38	45	user35	user35@example....	password35	admin	2024-11-24 18:27:30.281	ativo	NULL
39	46	user36	user36@example....	password36	funcionario	2024-11-24 18:27:30.281	ativo	NULL
40	47	user37	user37@example....	password37	gerente	2024-11-24 18:27:30.281	ativo	NULL
41	48	user38	user38@example....	password38	funcionario	2024-11-24 18:27:30.281	ativo	NULL
42	49	user39	user39@example....	password39	admin	2024-11-24 18:27:30.281	ativo	NULL
43	50	user40	user40@example....	password40	funcionario	2024-11-24 18:27:30.281	inativo	NULL
44	51	user41	user41@example....	password41	gerente	2024-11-24 18:27:30.281	ativo	NULL
45	52	user42	user42@example....	password42	funcionario	2024-11-24 18:27:30.281	ativo	NULL
46	53	user43	user43@example....	password43	admin	2024-11-24 18:27:30.281	ativo	NULL
47	54	user44	user44@example....	password44	funcionario	2024-11-24 18:27:30.281	ativo	NULL
48	55	user45	user45@example....	password45	gerente	2024-11-24 18:27:30.281	inativo	NULL
49	56	user46	user46@example....	password46	funcionario	2024-11-24 18:27:30.281	ativo	NULL
50	57	user47	user47@example....	password47	admin	2024-11-24 18:27:30.281	ativo	NULL
51	58	user48	user48@example....	password48	funcionario	2024-11-24 18:27:30.281	ativo	NULL
52	59	user49	user49@example....	password49	gerente	2024-11-24 18:27:30.281	ativo	NULL

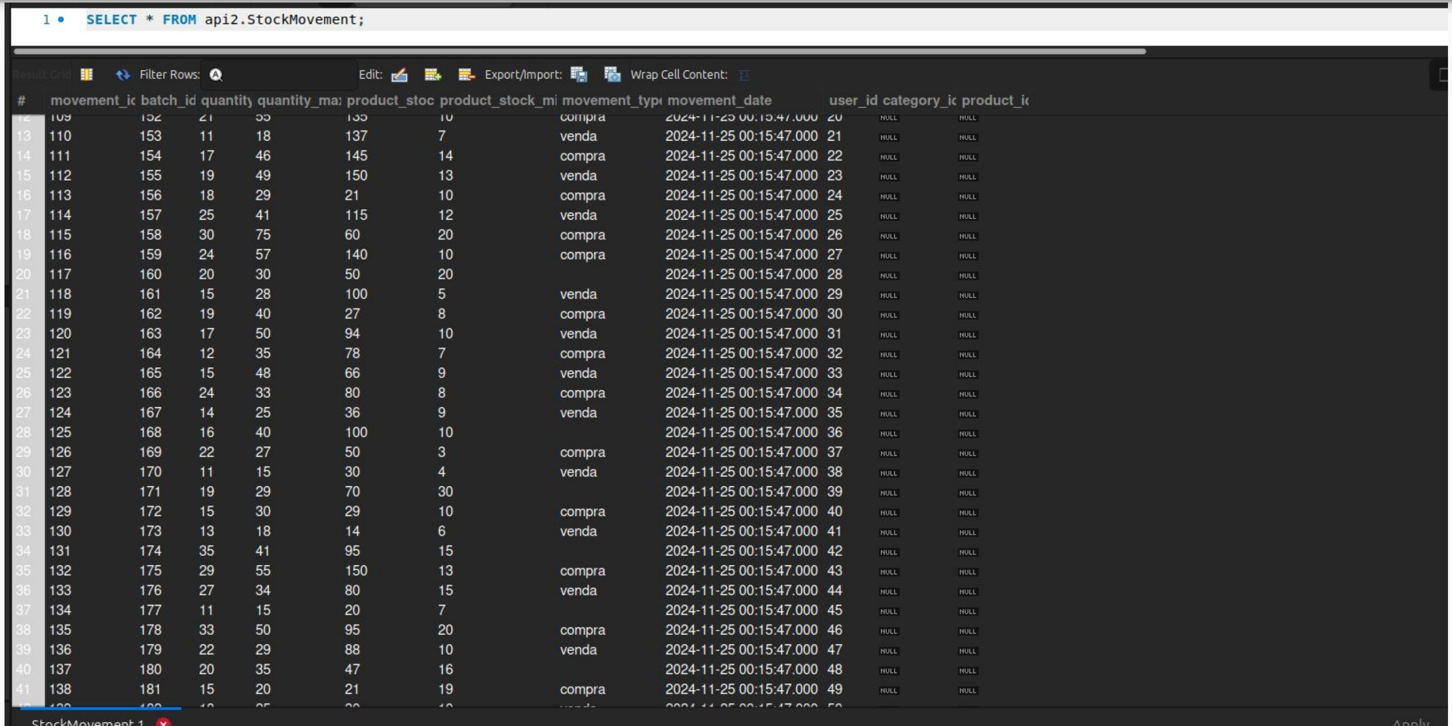
DML - INSERTS tabela Supplier



```
SELECT * FROM api2.Supplier;
```

#	supplier_ic	supplier_nam	contact_info	supplier_im	address	created_at
25	25	Supplier 22	contact22@example....	NULL	Endereço...	2024-11-24 18:34:32.187
26	26	Supplier 23	contact23@example....	NULL	Endereço...	2024-11-24 18:34:32.187
27	27	Supplier 24	contact24@example....	NULL	Endereço...	2024-11-24 18:34:32.187
28	28	Supplier 25	contact25@example....	NULL	Endereço...	2024-11-24 18:34:32.187
29	29	Supplier 26	contact26@example....	NULL	Endereço...	2024-11-24 18:34:32.187
30	30	Supplier 27	contact27@example....	NULL	Endereço...	2024-11-24 18:34:32.187
31	31	Supplier 28	contact28@example....	NULL	Endereço...	2024-11-24 18:34:32.187
32	32	Supplier 29	contact29@example....	NULL	Endereço...	2024-11-24 18:34:32.187
33	33	Supplier 30	contact30@example....	NULL	Endereço...	2024-11-24 18:34:32.187
34	34	Supplier 31	contact31@example....	NULL	Endereço...	2024-11-24 18:34:32.187
35	35	Supplier 32	contact32@example....	NULL	Endereço...	2024-11-24 18:34:32.187
36	36	Supplier 33	contact33@example....	NULL	Endereço...	2024-11-24 18:34:32.187
37	37	Supplier 34	contact34@example....	NULL	Endereço...	2024-11-24 18:34:32.187
38	38	Supplier 35	contact35@example....	NULL	Endereço...	2024-11-24 18:34:32.187
39	39	Supplier 36	contact36@example....	NULL	Endereço...	2024-11-24 18:34:32.187
40	40	Supplier 37	contact37@example....	NULL	Endereço...	2024-11-24 18:34:32.187
41	41	Supplier 38	contact38@example....	NULL	Endereço...	2024-11-24 18:34:32.187
42	42	Supplier 39	contact39@example....	NULL	Endereço...	2024-11-24 18:34:32.187
43	43	Supplier 40	contact40@example....	NULL	Endereço...	2024-11-24 18:34:32.187
44	44	Supplier 41	contact41@example....	NULL	Endereço...	2024-11-24 18:34:32.187
45	45	Supplier 42	contact42@example....	NULL	Endereço...	2024-11-24 18:34:32.187
46	46	Supplier 43	contact43@example....	NULL	Endereço...	2024-11-24 18:34:32.187
47	47	Supplier 44	contact44@example....	NULL	Endereço...	2024-11-24 18:34:32.187
48	48	Supplier 45	contact45@example....	NULL	Endereço...	2024-11-24 18:34:32.187
49	49	Supplier 46	contact46@example....	NULL	Endereço...	2024-11-24 18:34:32.187
50	50	Supplier 47	contact47@example....	NULL	Endereço...	2024-11-24 18:34:32.187
51	51	Supplier 48	contact48@example....	NULL	Endereço...	2024-11-24 18:34:32.187
52	52	Supplier 49	contact49@example....	NULL	Endereço...	2024-11-24 18:34:32.187
53	53	Supplier 50	contact50@example....	NULL	Endereço...	2024-11-24 18:34:32.187
*	NULL	NULL	NULL	NULL	NULL	NULL

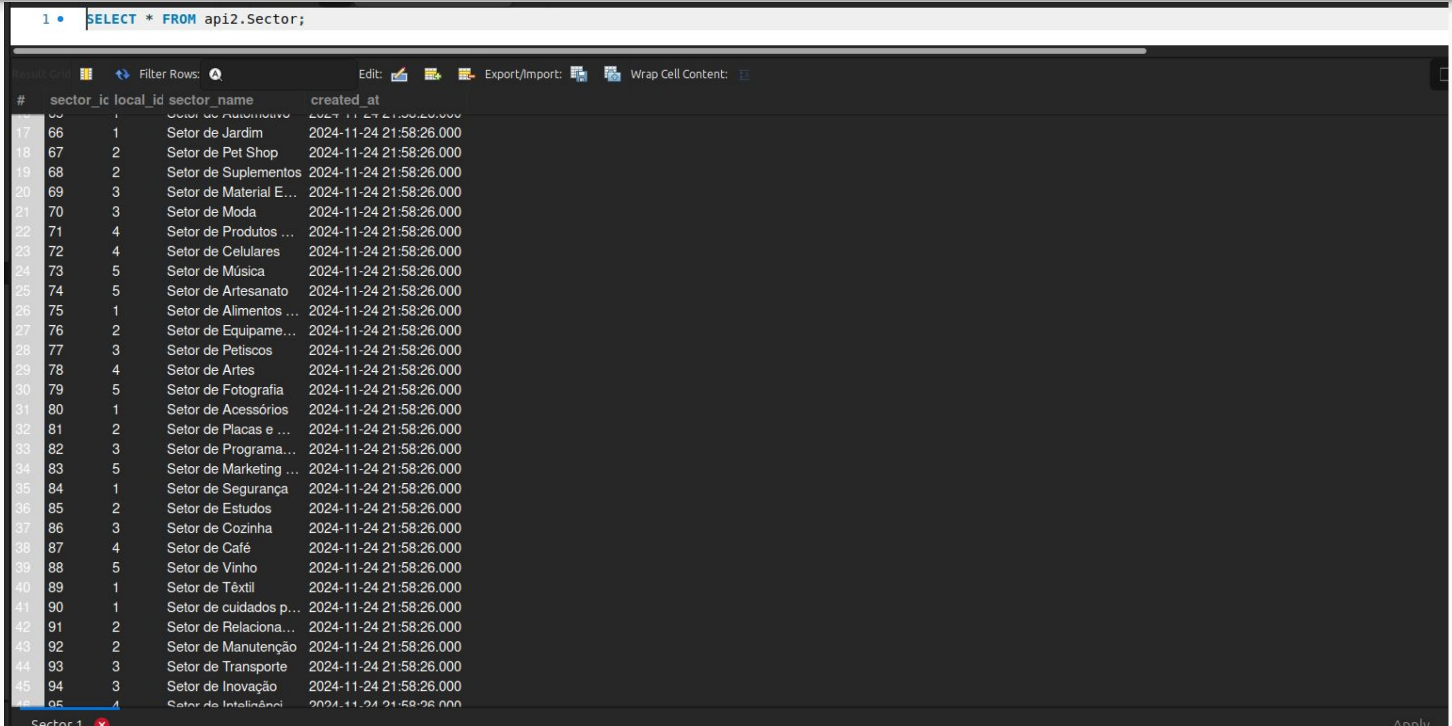
DML - INSERTS tabela StockMovement



```
SELECT * FROM api2.StockMovement;
```

#	movement_ic	batch_id	quantity	quantity_ma:	product_stoc	product_stock_mi	movement_typ	movement_date	user_id	category_ic	product_ic
12	109	152	21	55	135	10	compra	2024-11-25 00:15:47.000	20	NULL	NULL
13	110	153	11	18	137	7	venda	2024-11-25 00:15:47.000	21	NULL	NULL
14	111	154	17	46	145	14	compra	2024-11-25 00:15:47.000	22	NULL	NULL
15	112	155	19	49	150	13	venda	2024-11-25 00:15:47.000	23	NULL	NULL
16	113	156	18	29	21	10	compra	2024-11-25 00:15:47.000	24	NULL	NULL
17	114	157	25	41	115	12	venda	2024-11-25 00:15:47.000	25	NULL	NULL
18	115	158	30	75	60	20	compra	2024-11-25 00:15:47.000	26	NULL	NULL
19	116	159	24	57	140	10	compra	2024-11-25 00:15:47.000	27	NULL	NULL
20	117	160	20	30	50	20		2024-11-25 00:15:47.000	28	NULL	NULL
21	118	161	15	28	100	5	venda	2024-11-25 00:15:47.000	29	NULL	NULL
22	119	162	19	40	27	8	compra	2024-11-25 00:15:47.000	30	NULL	NULL
23	120	163	17	50	94	10	venda	2024-11-25 00:15:47.000	31	NULL	NULL
24	121	164	12	35	78	7	compra	2024-11-25 00:15:47.000	32	NULL	NULL
25	122	165	15	48	66	9	venda	2024-11-25 00:15:47.000	33	NULL	NULL
26	123	166	24	33	80	8	compra	2024-11-25 00:15:47.000	34	NULL	NULL
27	124	167	14	25	36	9	venda	2024-11-25 00:15:47.000	35	NULL	NULL
28	125	168	16	40	100	10		2024-11-25 00:15:47.000	36	NULL	NULL
29	126	169	22	27	50	3	compra	2024-11-25 00:15:47.000	37	NULL	NULL
30	127	170	11	15	30	4	venda	2024-11-25 00:15:47.000	38	NULL	NULL
31	128	171	19	29	70	30		2024-11-25 00:15:47.000	39	NULL	NULL
32	129	172	15	30	29	10	compra	2024-11-25 00:15:47.000	40	NULL	NULL
33	130	173	13	18	14	6	venda	2024-11-25 00:15:47.000	41	NULL	NULL
34	131	174	35	41	95	15		2024-11-25 00:15:47.000	42	NULL	NULL
35	132	175	29	55	150	13	compra	2024-11-25 00:15:47.000	43	NULL	NULL
36	133	176	27	34	80	15	venda	2024-11-25 00:15:47.000	44	NULL	NULL
37	134	177	11	15	20	7		2024-11-25 00:15:47.000	45	NULL	NULL
38	135	178	33	50	95	20	compra	2024-11-25 00:15:47.000	46	NULL	NULL
39	136	179	22	29	88	10	venda	2024-11-25 00:15:47.000	47	NULL	NULL
40	137	180	20	35	47	16		2024-11-25 00:15:47.000	48	NULL	NULL
41	138	181	15	20	21	19	compra	2024-11-25 00:15:47.000	49	NULL	NULL

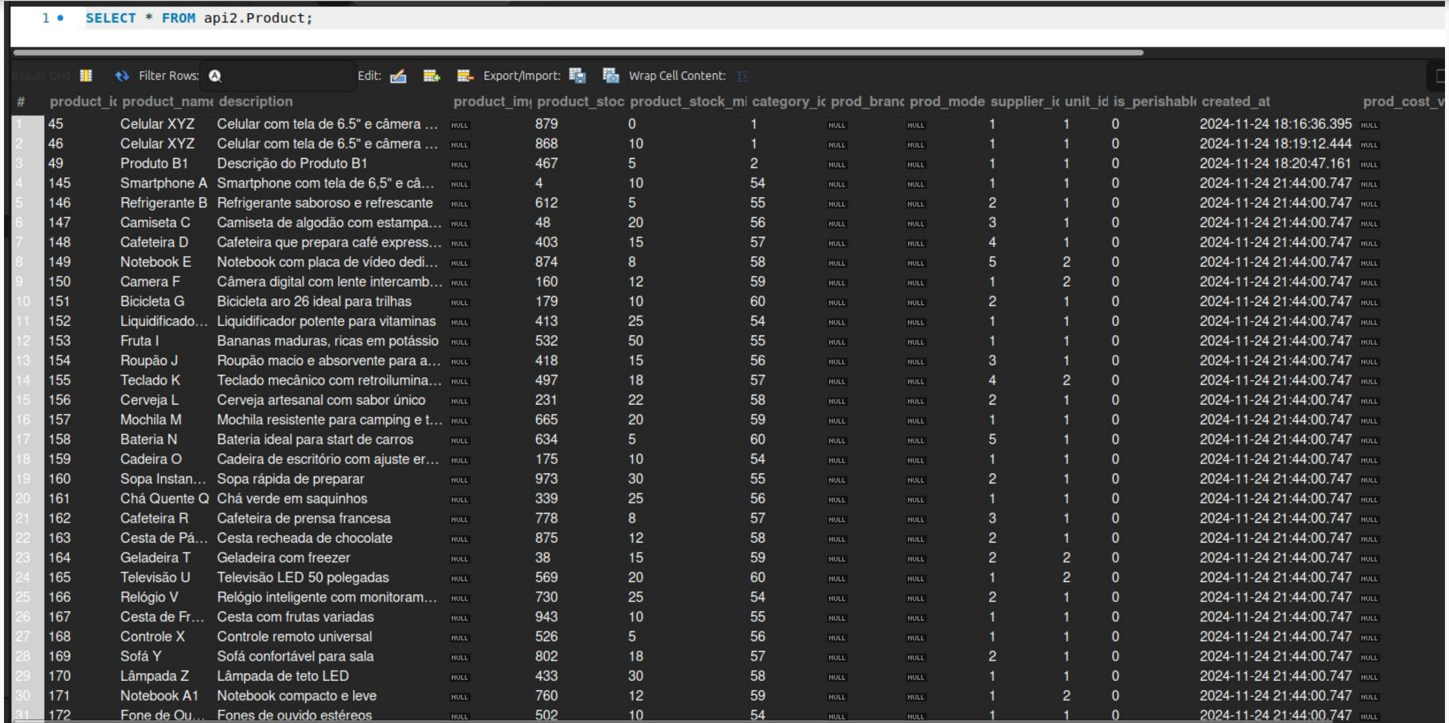
DML - INSERTS tabela Sector



```
SELECT * FROM api2.Sector;
```

#	sector_ic	local_id	sector_name	created_at
16	65	1	Setor de Automotivo	2024-11-24 21:58:26.000
17	66	1	Setor de Jardim	2024-11-24 21:58:26.000
18	67	2	Setor de Pet Shop	2024-11-24 21:58:26.000
19	68	2	Setor de Suplementos	2024-11-24 21:58:26.000
20	69	3	Setor de Material E…	2024-11-24 21:58:26.000
21	70	3	Setor de Moda	2024-11-24 21:58:26.000
22	71	4	Setor de Produtos …	2024-11-24 21:58:26.000
23	72	4	Setor de Celulares	2024-11-24 21:58:26.000
24	73	5	Setor de Música	2024-11-24 21:58:26.000
25	74	5	Setor de Artesanato	2024-11-24 21:58:26.000
26	75	1	Setor de Alimentos …	2024-11-24 21:58:26.000
27	76	2	Setor de Equipame…	2024-11-24 21:58:26.000
28	77	3	Setor de Petiscos	2024-11-24 21:58:26.000
29	78	4	Setor de Artes	2024-11-24 21:58:26.000
30	79	5	Setor de Fotografia	2024-11-24 21:58:26.000
31	80	1	Setor de Acessórios	2024-11-24 21:58:26.000
32	81	2	Setor de Placas e …	2024-11-24 21:58:26.000
33	82	3	Setor de Programa…	2024-11-24 21:58:26.000
34	83	5	Setor de Marketing …	2024-11-24 21:58:26.000
35	84	1	Setor de Segurança	2024-11-24 21:58:26.000
36	85	2	Setor de Estudos	2024-11-24 21:58:26.000
37	86	3	Setor de Cozinha	2024-11-24 21:58:26.000
38	87	4	Setor de Café	2024-11-24 21:58:26.000
39	88	5	Setor de Vinho	2024-11-24 21:58:26.000
40	89	1	Setor de Têxtil	2024-11-24 21:58:26.000
41	90	1	Setor de cuidados p…	2024-11-24 21:58:26.000
42	91	2	Setor de Relaciona…	2024-11-24 21:58:26.000
43	92	2	Setor de Manutenção	2024-11-24 21:58:26.000
44	93	3	Setor de Transporte	2024-11-24 21:58:26.000
45	94	3	Setor de Inovação	2024-11-24 21:58:26.000
46	95	4	Setor de Inteligênci	2024-11-24 21:58:26.000

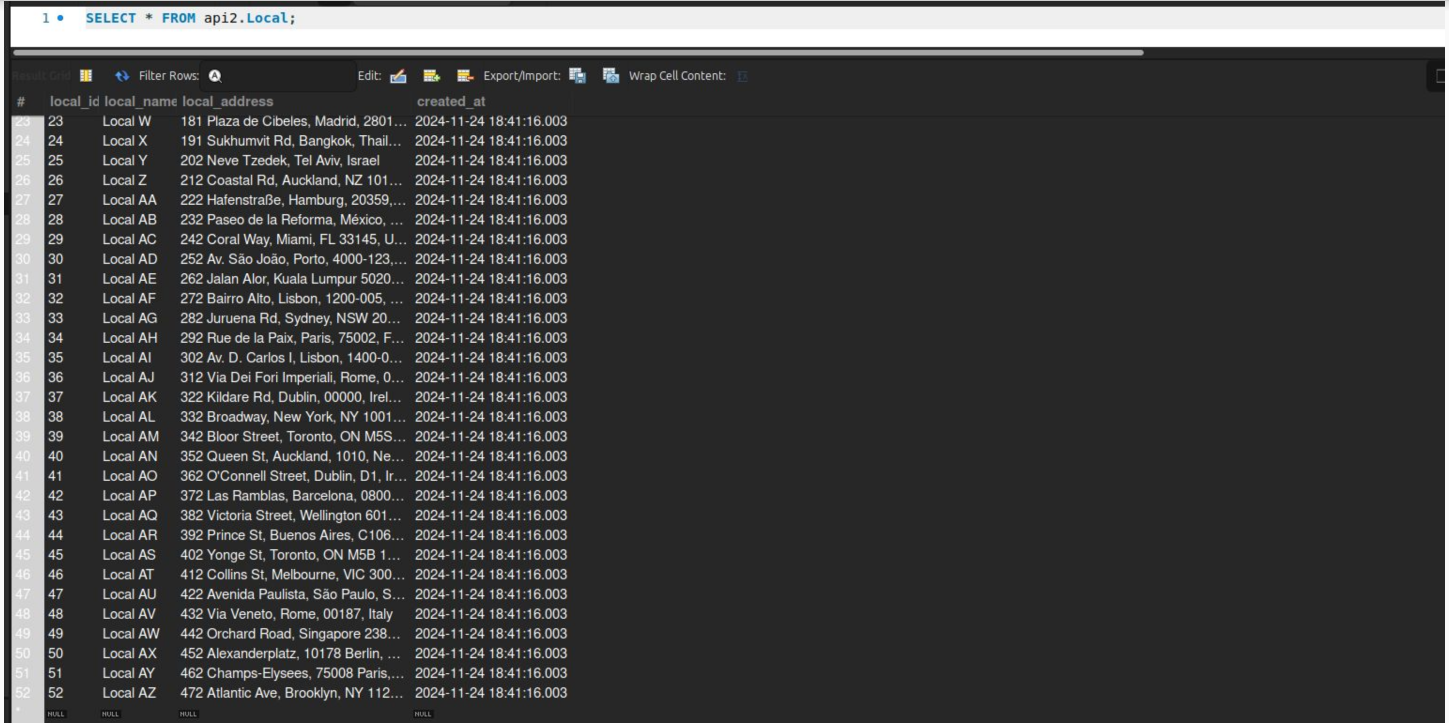
DML - INSERTS tabela Product



```
1 • SELECT * FROM api2.Product;
```

#	product_ic	product_name	description	product_im	product_stoc	product_stock_mi	category_ic	prod_branc	prod_mode	supplier_ic	unit_id	is_perishabl	created_at	prod_cost_v
1	45	Celular XYZ	Celular com tela de 6.5" e câmera ...	NULL	879	0	1	NULL	NULL	1	1	0	2024-11-24 18:16:36.395	NULL
2	46	Celular XYZ	Celular com tela de 6.5" e câmera ...	NULL	868	10	1	NULL	NULL	1	1	0	2024-11-24 18:19:12.444	NULL
3	49	Produto B1	Descrição do Produto B1	NULL	467	5	2	NULL	NULL	1	1	0	2024-11-24 18:20:47.161	NULL
4	145	Smartphone A	Smartphone com tela de 6,5" e câ...	NULL	4	10	54	NULL	NULL	1	1	0	2024-11-24 21:44:00.747	NULL
5	146	Refrigerante B	Refrigerante saboroso e refrescante	NULL	612	5	55	NULL	NULL	2	1	0	2024-11-24 21:44:00.747	NULL
6	147	Camiseta C	Camiseta de algodão com estampa...	NULL	48	20	56	NULL	NULL	3	1	0	2024-11-24 21:44:00.747	NULL
7	148	Cafeteira D	Cafeteira que prepara café express...	NULL	403	15	57	NULL	NULL	4	1	0	2024-11-24 21:44:00.747	NULL
8	149	Notebook E	Notebook com placa de vídeo dedi...	NULL	874	8	58	NULL	NULL	5	2	0	2024-11-24 21:44:00.747	NULL
9	150	Camera F	Câmera digital com lente intercamb...	NULL	160	12	59	NULL	NULL	1	2	0	2024-11-24 21:44:00.747	NULL
10	151	Bicicleta G	Bicicleta aro 26 ideal para trilhas	NULL	179	10	60	NULL	NULL	2	1	0	2024-11-24 21:44:00.747	NULL
11	152	Liquidificado...	Liquidificador potente para vitaminas	NULL	413	25	54	NULL	NULL	1	1	0	2024-11-24 21:44:00.747	NULL
12	153	Fruta I	Bananas maduras, ricas em potássio	NULL	532	50	55	NULL	NULL	1	1	0	2024-11-24 21:44:00.747	NULL
13	154	Roupão J	Roupão macio e absorvente para a...	NULL	418	15	56	NULL	NULL	3	1	0	2024-11-24 21:44:00.747	NULL
14	155	Teclado K	Teclado mecânico com retroilumina...	NULL	497	18	57	NULL	NULL	4	2	0	2024-11-24 21:44:00.747	NULL
15	156	Cerveja L	Cerveja artesanal com sabor único	NULL	231	22	58	NULL	NULL	2	1	0	2024-11-24 21:44:00.747	NULL
16	157	Mochila M	Mochila resistente para camping e t...	NULL	665	20	59	NULL	NULL	1	1	0	2024-11-24 21:44:00.747	NULL
17	158	Bateria N	Bateria ideal para start de carros	NULL	634	5	60	NULL	NULL	5	1	0	2024-11-24 21:44:00.747	NULL
18	159	Cadeira O	Cadeira de escritório com ajuste er...	NULL	175	10	54	NULL	NULL	1	1	0	2024-11-24 21:44:00.747	NULL
19	160	Sopa Instan...	Sopa rápida de preparar	NULL	973	30	55	NULL	NULL	2	1	0	2024-11-24 21:44:00.747	NULL
20	161	Chá Quente Q	Chá verde em saquinhos	NULL	339	25	56	NULL	NULL	1	1	0	2024-11-24 21:44:00.747	NULL
21	162	Cafeteira R	Cafeteira de prensa francesa	NULL	778	8	57	NULL	NULL	3	1	0	2024-11-24 21:44:00.747	NULL
22	163	Cesta de Pá...	Cesta recheada de chocolate	NULL	875	12	58	NULL	NULL	2	1	0	2024-11-24 21:44:00.747	NULL
23	164	Geladeira T	Geladeira com freezer	NULL	38	15	59	NULL	NULL	2	2	0	2024-11-24 21:44:00.747	NULL
24	165	Televisão U	Televisão LED 50 polegadas	NULL	569	20	60	NULL	NULL	1	2	0	2024-11-24 21:44:00.747	NULL
25	166	Relógio V	Relógio inteligente com monitoram...	NULL	730	25	54	NULL	NULL	2	1	0	2024-11-24 21:44:00.747	NULL
26	167	Cesta de Fr...	Cesta com frutas variadas	NULL	943	10	55	NULL	NULL	1	1	0	2024-11-24 21:44:00.747	NULL
27	168	Controle X	Controle remoto universal	NULL	526	5	56	NULL	NULL	1	1	0	2024-11-24 21:44:00.747	NULL
28	169	Sofá Y	Sofá confortável para sala	NULL	802	18	57	NULL	NULL	2	1	0	2024-11-24 21:44:00.747	NULL
29	170	Lâmpada Z	Lâmpada de teto LED	NULL	433	30	58	NULL	NULL	1	1	0	2024-11-24 21:44:00.747	NULL
30	171	Notebook A1	Notebook compacto e leve	NULL	760	12	59	NULL	NULL	1	2	0	2024-11-24 21:44:00.747	NULL
31	172	Fone de Ou...	Fones de ouvido estéreos	NULL	502	10	54	NULL	NULL	1	1	0	2024-11-24 21:44:00.747	NULL

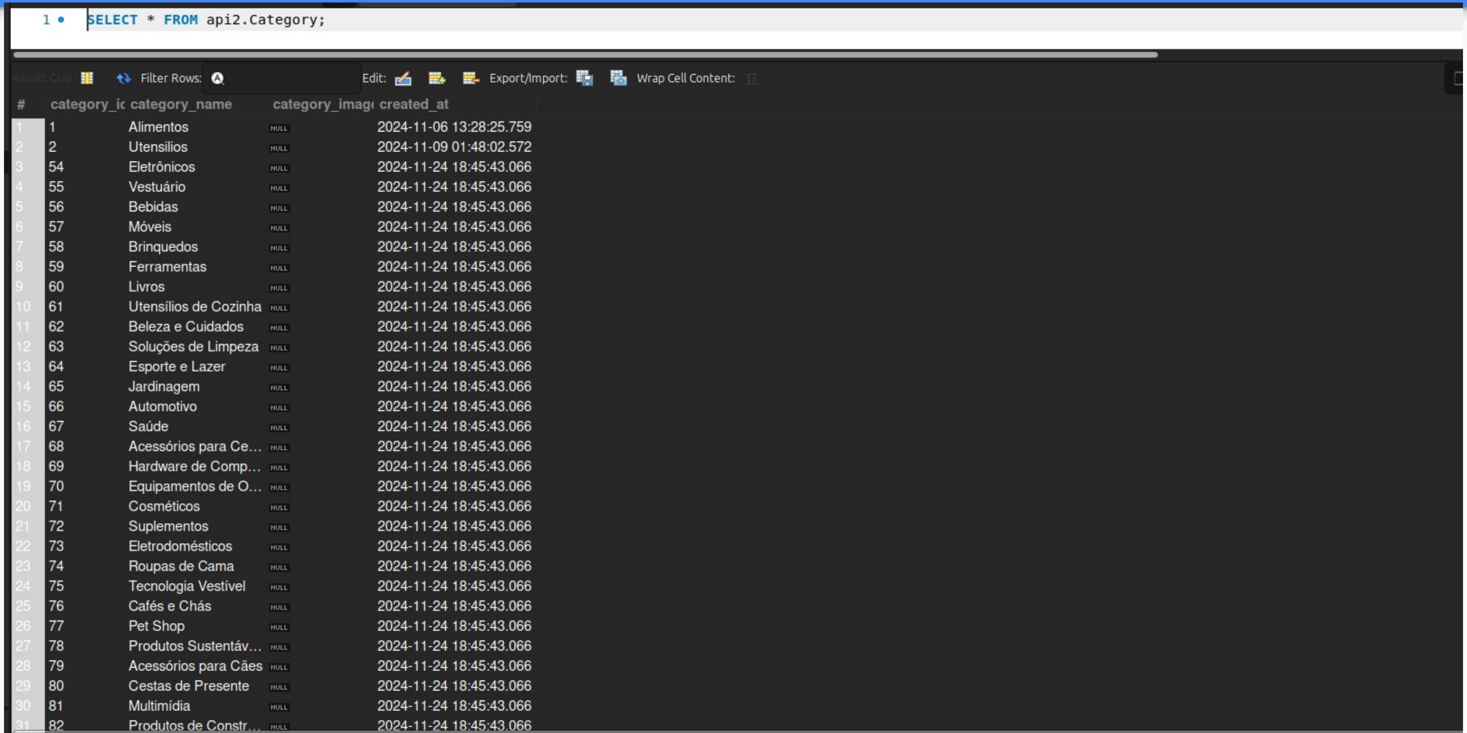
DML - INSERTS tabela Local



```sql
SELECT * FROM api2.Local;
```

#	local_id	local_name	local_address	created_at
23	23	Local W	181 Plaza de Cibeles, Madrid, 2801…	2024-11-24 18:41:16.003
24	24	Local X	191 Sukhumvit Rd, Bangkok, Thail…	2024-11-24 18:41:16.003
25	25	Local Y	202 Neve Tzedek, Tel Aviv, Israel	2024-11-24 18:41:16.003
26	26	Local Z	212 Coastal Rd, Auckland, NZ 101…	2024-11-24 18:41:16.003
27	27	Local AA	222 Hafenstraße, Hamburg, 20359,…	2024-11-24 18:41:16.003
28	28	Local AB	232 Paseo de la Reforma, México, …	2024-11-24 18:41:16.003
29	29	Local AC	242 Coral Way, Miami, FL 33145, U…	2024-11-24 18:41:16.003
30	30	Local AD	252 Av. São João, Porto, 4000-123,…	2024-11-24 18:41:16.003
31	31	Local AE	262 Jalan Alor, Kuala Lumpur 5020…	2024-11-24 18:41:16.003
32	32	Local AF	272 Bairro Alto, Lisbon, 1200-005, …	2024-11-24 18:41:16.003
33	33	Local AG	282 Juruena Rd, Sydney, NSW 20…	2024-11-24 18:41:16.003
34	34	Local AH	292 Rue de la Paix, Paris, 75002, F…	2024-11-24 18:41:16.003
35	35	Local AI	302 Av. D. Carlos I, Lisbon, 1400-0…	2024-11-24 18:41:16.003
36	36	Local AJ	312 Via Dei Fori Imperiali, Rome, 0…	2024-11-24 18:41:16.003
37	37	Local AK	322 Kildare Rd, Dublin, 00000, Irel…	2024-11-24 18:41:16.003
38	38	Local AL	332 Broadway, New York, NY 1001…	2024-11-24 18:41:16.003
39	39	Local AM	342 Bloor Street, Toronto, ON M5S…	2024-11-24 18:41:16.003
40	40	Local AN	352 Queen St, Auckland, 1010, Ne…	2024-11-24 18:41:16.003
41	41	Local AO	362 O'Connell Street, Dublin, D1, Ir…	2024-11-24 18:41:16.003
42	42	Local AP	372 Las Ramblas, Barcelona, 0800…	2024-11-24 18:41:16.003
43	43	Local AQ	382 Victoria Street, Wellington 601…	2024-11-24 18:41:16.003
44	44	Local AR	392 Prince St, Buenos Aires, C106…	2024-11-24 18:41:16.003
45	45	Local AS	402 Yonge St, Toronto, ON M5B 1…	2024-11-24 18:41:16.003
46	46	Local AT	412 Collins St, Melbourne, VIC 300…	2024-11-24 18:41:16.003
47	47	Local AU	422 Avenida Paulista, São Paulo, S…	2024-11-24 18:41:16.003
48	48	Local AV	432 Via Veneto, Rome, 00187, Italy	2024-11-24 18:41:16.003
49	49	Local AW	442 Orchard Road, Singapore 238…	2024-11-24 18:41:16.003
50	50	Local AX	452 Alexanderplatz, 10178 Berlin, …	2024-11-24 18:41:16.003
51	51	Local AY	462 Champs-Elysees, 75008 Paris,…	2024-11-24 18:41:16.003
52	52	Local AZ	472 Atlantic Ave, Brooklyn, NY 112…	2024-11-24 18:41:16.003
*	NULL	NULL	NULL	NULL

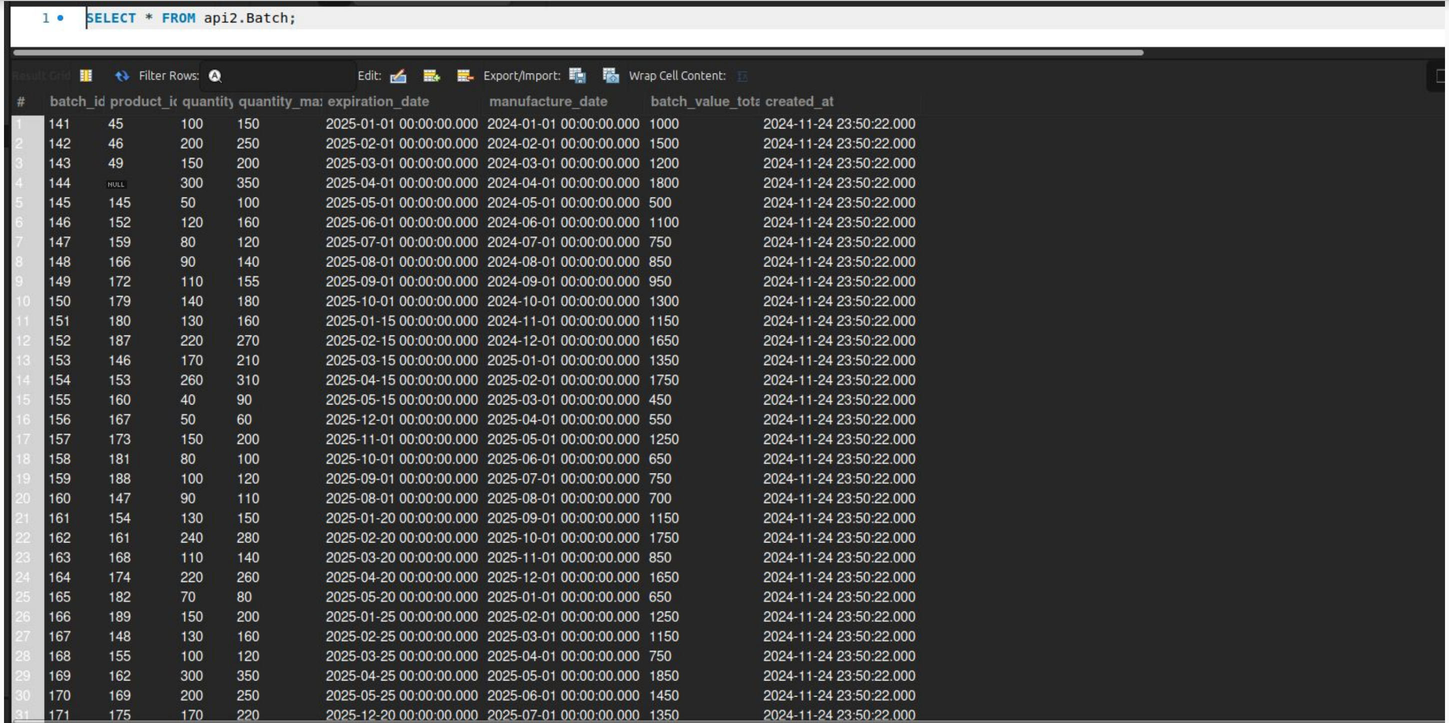
DML - INSERTS tabela category



```sql
SELECT * FROM api2.Category;
```

#	category_ic	category_name	category_imag	created_at
1	1	Alimentos	NULL	2024-11-06 13:28:25.759
2	2	Utensilios	NULL	2024-11-09 01:48:02.572
3	54	Eletrônicos	NULL	2024-11-24 18:45:43.066
4	55	Vestuário	NULL	2024-11-24 18:45:43.066
5	56	Bebidas	NULL	2024-11-24 18:45:43.066
6	57	Móveis	NULL	2024-11-24 18:45:43.066
7	58	Brinquedos	NULL	2024-11-24 18:45:43.066
8	59	Ferramentas	NULL	2024-11-24 18:45:43.066
9	60	Livros	NULL	2024-11-24 18:45:43.066
10	61	Utensílios de Cozinha	NULL	2024-11-24 18:45:43.066
11	62	Beleza e Cuidados	NULL	2024-11-24 18:45:43.066
12	63	Soluções de Limpeza	NULL	2024-11-24 18:45:43.066
13	64	Esporte e Lazer	NULL	2024-11-24 18:45:43.066
14	65	Jardinagem	NULL	2024-11-24 18:45:43.066
15	66	Automotivo	NULL	2024-11-24 18:45:43.066
16	67	Saúde	NULL	2024-11-24 18:45:43.066
17	68	Acessórios para Ce...	NULL	2024-11-24 18:45:43.066
18	69	Hardware de Comp...	NULL	2024-11-24 18:45:43.066
19	70	Equipamentos de O...	NULL	2024-11-24 18:45:43.066
20	71	Cosméticos	NULL	2024-11-24 18:45:43.066
21	72	Suplementos	NULL	2024-11-24 18:45:43.066
22	73	Eletrodomésticos	NULL	2024-11-24 18:45:43.066
23	74	Roupas de Cama	NULL	2024-11-24 18:45:43.066
24	75	Tecnologia Vestível	NULL	2024-11-24 18:45:43.066
25	76	Cafés e Chás	NULL	2024-11-24 18:45:43.066
26	77	Pet Shop	NULL	2024-11-24 18:45:43.066
27	78	Produtos Sustentáv...	NULL	2024-11-24 18:45:43.066
28	79	Acessórios para Cães	NULL	2024-11-24 18:45:43.066
29	80	Cestas de Presente	NULL	2024-11-24 18:45:43.066
30	81	Multimídia	NULL	2024-11-24 18:45:43.066
31	82	Produtos de Constr...	NULL	2024-11-24 18:45:43.066

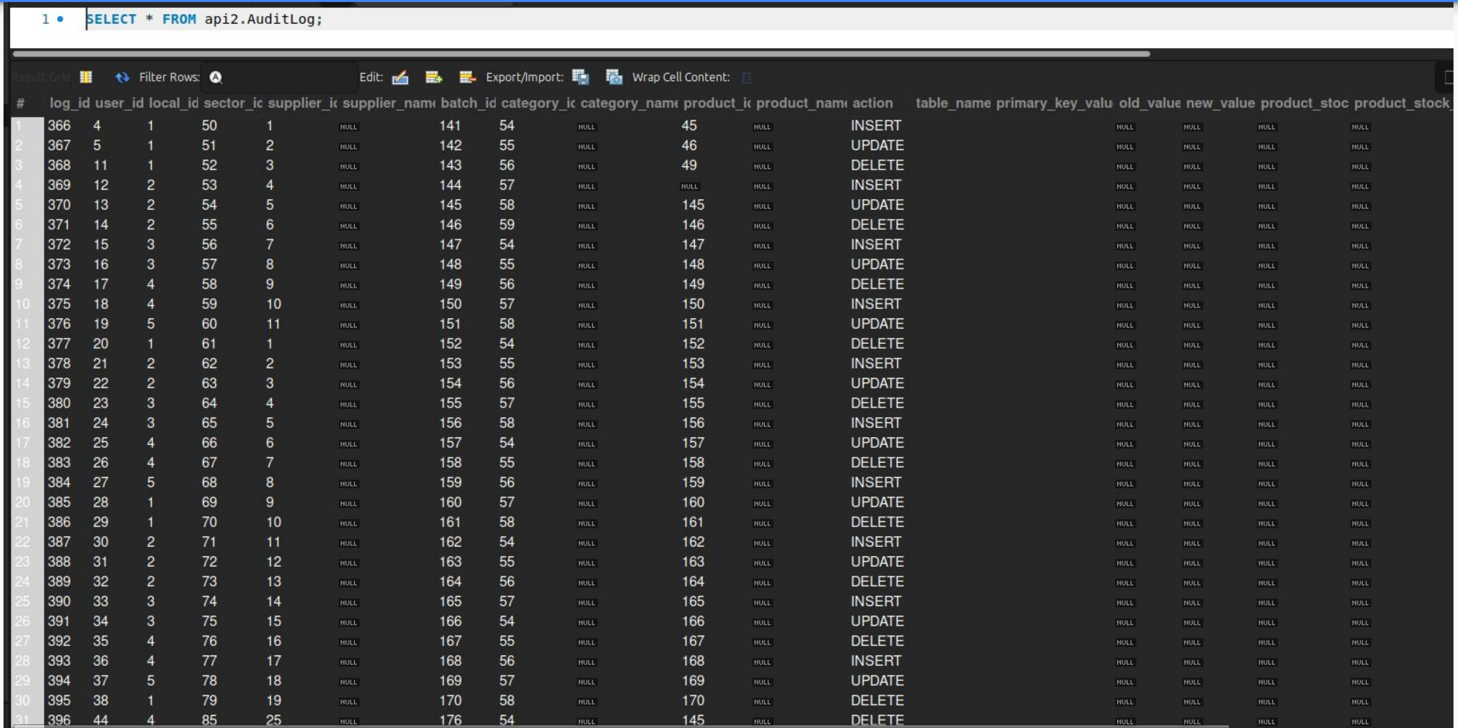
DML - INSERTS tabela Batch



```
SELECT * FROM api2.Batch;
```

#	batch_id	product_ic	quantity	quantity_ma:	expiration_date	manufacture_date	batch_value_tota	created_at
1	141	45	100	150	2025-01-01 00:00:00.000	2024-01-01 00:00:00.000	1000	2024-11-24 23:50:22.000
2	142	46	200	250	2025-02-01 00:00:00.000	2024-02-01 00:00:00.000	1500	2024-11-24 23:50:22.000
3	143	49	150	200	2025-03-01 00:00:00.000	2024-03-01 00:00:00.000	1200	2024-11-24 23:50:22.000
4	144	NULL	300	350	2025-04-01 00:00:00.000	2024-04-01 00:00:00.000	1800	2024-11-24 23:50:22.000
5	145	145	50	100	2025-05-01 00:00:00.000	2024-05-01 00:00:00.000	500	2024-11-24 23:50:22.000
6	146	152	120	160	2025-06-01 00:00:00.000	2024-06-01 00:00:00.000	1100	2024-11-24 23:50:22.000
7	147	159	80	120	2025-07-01 00:00:00.000	2024-07-01 00:00:00.000	750	2024-11-24 23:50:22.000
8	148	166	90	140	2025-08-01 00:00:00.000	2024-08-01 00:00:00.000	850	2024-11-24 23:50:22.000
9	149	172	110	155	2025-09-01 00:00:00.000	2024-09-01 00:00:00.000	950	2024-11-24 23:50:22.000
10	150	179	140	180	2025-10-01 00:00:00.000	2024-10-01 00:00:00.000	1300	2024-11-24 23:50:22.000
11	151	180	130	160	2025-01-15 00:00:00.000	2024-11-01 00:00:00.000	1150	2024-11-24 23:50:22.000
12	152	187	220	270	2025-02-15 00:00:00.000	2024-12-01 00:00:00.000	1650	2024-11-24 23:50:22.000
13	153	146	170	210	2025-03-15 00:00:00.000	2025-01-01 00:00:00.000	1350	2024-11-24 23:50:22.000
14	154	153	260	310	2025-04-15 00:00:00.000	2025-02-01 00:00:00.000	1750	2024-11-24 23:50:22.000
15	155	160	40	90	2025-05-15 00:00:00.000	2025-03-01 00:00:00.000	450	2024-11-24 23:50:22.000
16	156	167	50	60	2025-12-01 00:00:00.000	2025-04-01 00:00:00.000	550	2024-11-24 23:50:22.000
17	157	173	150	200	2025-11-01 00:00:00.000	2025-05-01 00:00:00.000	1250	2024-11-24 23:50:22.000
18	158	181	80	100	2025-10-01 00:00:00.000	2025-06-01 00:00:00.000	650	2024-11-24 23:50:22.000
19	159	188	100	120	2025-09-01 00:00:00.000	2025-07-01 00:00:00.000	750	2024-11-24 23:50:22.000
20	160	147	90	110	2025-08-01 00:00:00.000	2025-08-01 00:00:00.000	700	2024-11-24 23:50:22.000
21	161	154	130	150	2025-01-20 00:00:00.000	2025-09-01 00:00:00.000	1150	2024-11-24 23:50:22.000
22	162	161	240	280	2025-02-20 00:00:00.000	2025-10-01 00:00:00.000	1750	2024-11-24 23:50:22.000
23	163	168	110	140	2025-03-20 00:00:00.000	2025-11-01 00:00:00.000	850	2024-11-24 23:50:22.000
24	164	174	220	260	2025-04-20 00:00:00.000	2025-12-01 00:00:00.000	1650	2024-11-24 23:50:22.000
25	165	182	70	80	2025-05-20 00:00:00.000	2025-01-01 00:00:00.000	650	2024-11-24 23:50:22.000
26	166	189	150	200	2025-01-25 00:00:00.000	2025-02-01 00:00:00.000	1250	2024-11-24 23:50:22.000
27	167	148	130	160	2025-02-25 00:00:00.000	2025-03-01 00:00:00.000	1150	2024-11-24 23:50:22.000
28	168	155	100	120	2025-03-25 00:00:00.000	2025-04-01 00:00:00.000	750	2024-11-24 23:50:22.000
29	169	162	300	350	2025-04-25 00:00:00.000	2025-05-01 00:00:00.000	1850	2024-11-24 23:50:22.000
30	170	169	200	250	2025-05-25 00:00:00.000	2025-06-01 00:00:00.000	1450	2024-11-24 23:50:22.000
31	171	175	170	220	2025-12-20 00:00:00.000	2025-07-01 00:00:00.000	1350	2024-11-24 23:50:22.000

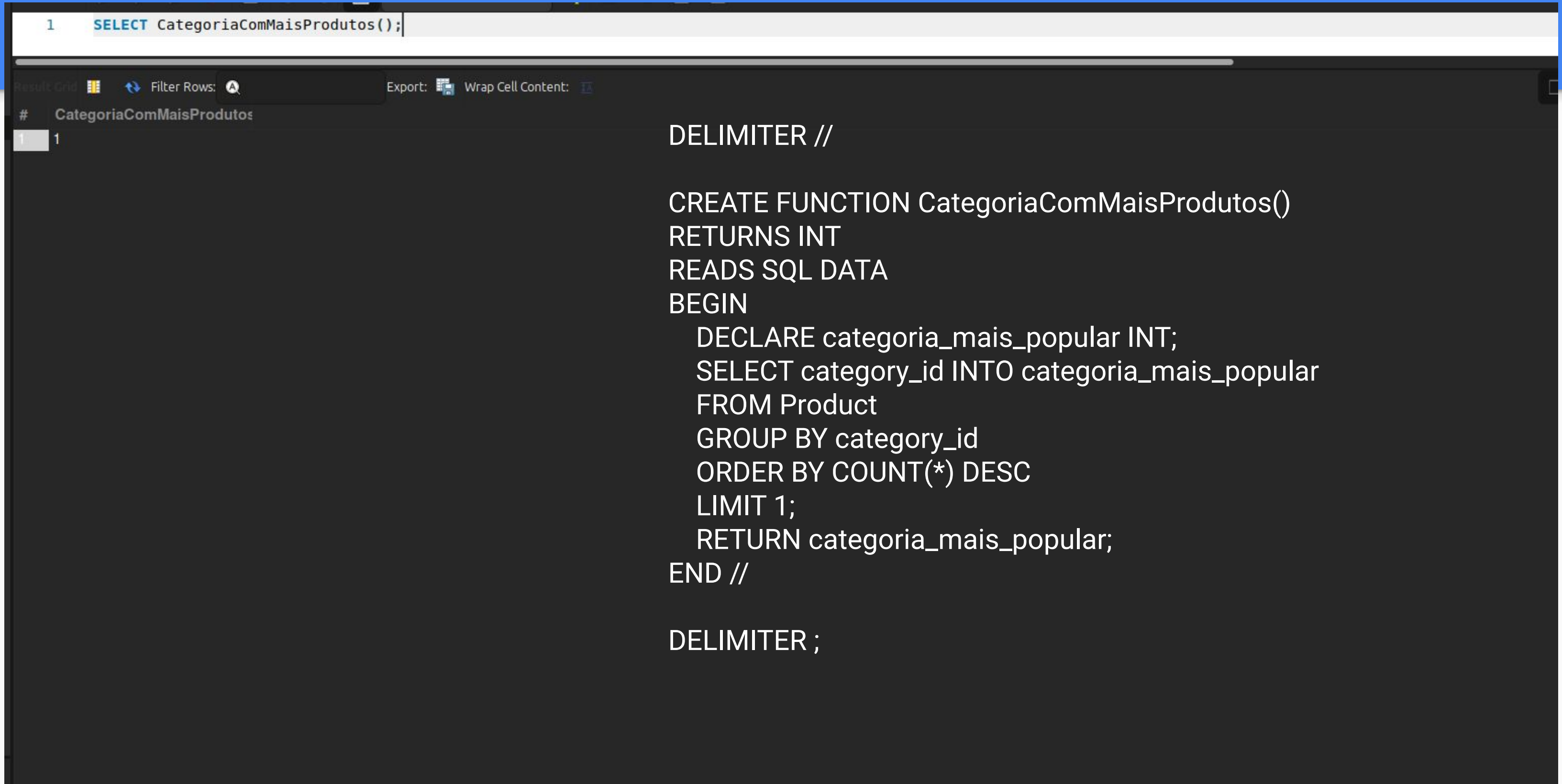
DML - INSERTS tabela AuditLog



```
1 • SELECT * FROM api2.AuditLog;
```

#	log_id	user_id	local_id	sector_ic	supplier_ic	supplier_nam	batch_id	category_ic	category_nam	product_ic	product_nam	action	table_name	primary_key_valu	old_value	new_value	product_stoc	product_stock_
1	366	4	1	50	1	NULL	141	54	NULL	45	NULL	INSERT			NULL	NULL	NULL	NULL
2	367	5	1	51	2	NULL	142	55	NULL	46	NULL	UPDATE			NULL	NULL	NULL	NULL
3	368	11	1	52	3	NULL	143	56	NULL	49	NULL	DELETE			NULL	NULL	NULL	NULL
4	369	12	2	53	4	NULL	144	57	NULL	NULL	NULL	INSERT			NULL	NULL	NULL	NULL
5	370	13	2	54	5	NULL	145	58	NULL	145	NULL	UPDATE			NULL	NULL	NULL	NULL
6	371	14	2	55	6	NULL	146	59	NULL	146	NULL	DELETE			NULL	NULL	NULL	NULL
7	372	15	3	56	7	NULL	147	54	NULL	147	NULL	INSERT			NULL	NULL	NULL	NULL
8	373	16	3	57	8	NULL	148	55	NULL	148	NULL	UPDATE			NULL	NULL	NULL	NULL
9	374	17	4	58	9	NULL	149	56	NULL	149	NULL	DELETE			NULL	NULL	NULL	NULL
10	375	18	4	59	10	NULL	150	57	NULL	150	NULL	INSERT			NULL	NULL	NULL	NULL
11	376	19	5	60	11	NULL	151	58	NULL	151	NULL	UPDATE			NULL	NULL	NULL	NULL
12	377	20	1	61	1	NULL	152	54	NULL	152	NULL	DELETE			NULL	NULL	NULL	NULL
13	378	21	2	62	2	NULL	153	55	NULL	153	NULL	INSERT			NULL	NULL	NULL	NULL
14	379	22	2	63	3	NULL	154	56	NULL	154	NULL	UPDATE			NULL	NULL	NULL	NULL
15	380	23	3	64	4	NULL	155	57	NULL	155	NULL	DELETE			NULL	NULL	NULL	NULL
16	381	24	3	65	5	NULL	156	58	NULL	156	NULL	INSERT			NULL	NULL	NULL	NULL
17	382	25	4	66	6	NULL	157	54	NULL	157	NULL	UPDATE			NULL	NULL	NULL	NULL
18	383	26	4	67	7	NULL	158	55	NULL	158	NULL	DELETE			NULL	NULL	NULL	NULL
19	384	27	5	68	8	NULL	159	56	NULL	159	NULL	INSERT			NULL	NULL	NULL	NULL
20	385	28	1	69	9	NULL	160	57	NULL	160	NULL	UPDATE			NULL	NULL	NULL	NULL
21	386	29	1	70	10	NULL	161	58	NULL	161	NULL	DELETE			NULL	NULL	NULL	NULL
22	387	30	2	71	11	NULL	162	54	NULL	162	NULL	INSERT			NULL	NULL	NULL	NULL
23	388	31	2	72	12	NULL	163	55	NULL	163	NULL	UPDATE			NULL	NULL	NULL	NULL
24	389	32	2	73	13	NULL	164	56	NULL	164	NULL	DELETE			NULL	NULL	NULL	NULL
25	390	33	3	74	14	NULL	165	57	NULL	165	NULL	INSERT			NULL	NULL	NULL	NULL
26	391	34	3	75	15	NULL	166	54	NULL	166	NULL	UPDATE			NULL	NULL	NULL	NULL
27	392	35	4	76	16	NULL	167	55	NULL	167	NULL	DELETE			NULL	NULL	NULL	NULL
28	393	36	4	77	17	NULL	168	56	NULL	168	NULL	INSERT			NULL	NULL	NULL	NULL
29	394	37	5	78	18	NULL	169	57	NULL	169	NULL	UPDATE			NULL	NULL	NULL	NULL
30	395	38	1	79	19	NULL	170	58	NULL	170	NULL	DELETE			NULL	NULL	NULL	NULL
31	396	44	4	85	25	NULL	176	54	NULL	145	NULL	DELETE			NULL	NULL	NULL	NULL

Functions - CategoriaComMaisProdutos()

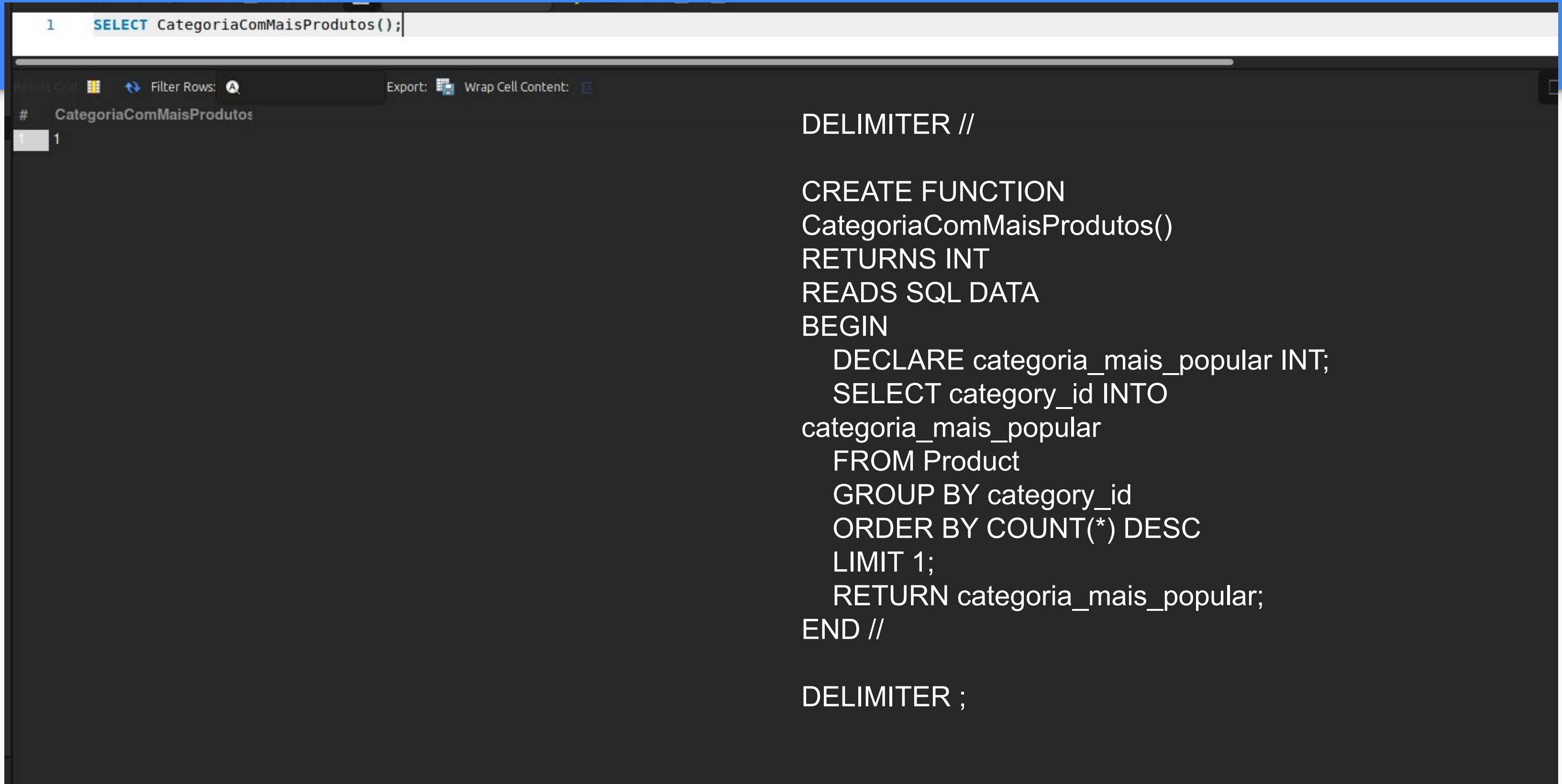


```
SELECT CategoriaComMaisProdutos();
```

```
#   CategoriaComMaisProdutos
    1
```

```
DELIMITER //

CREATE FUNCTION CategoriaComMaisProdutos()
RETURNS INT
READS SQL DATA
BEGIN
    DECLARE categoria_mais_popular INT;
    SELECT category_id INTO categoria_mais_popular
    FROM Product
    GROUP BY category_id
    ORDER BY COUNT(*) DESC
    LIMIT 1;
    RETURN categoria_mais_popular;
END //

DELIMITER ;
```

Functions - CategoriaComMaisProdutos()

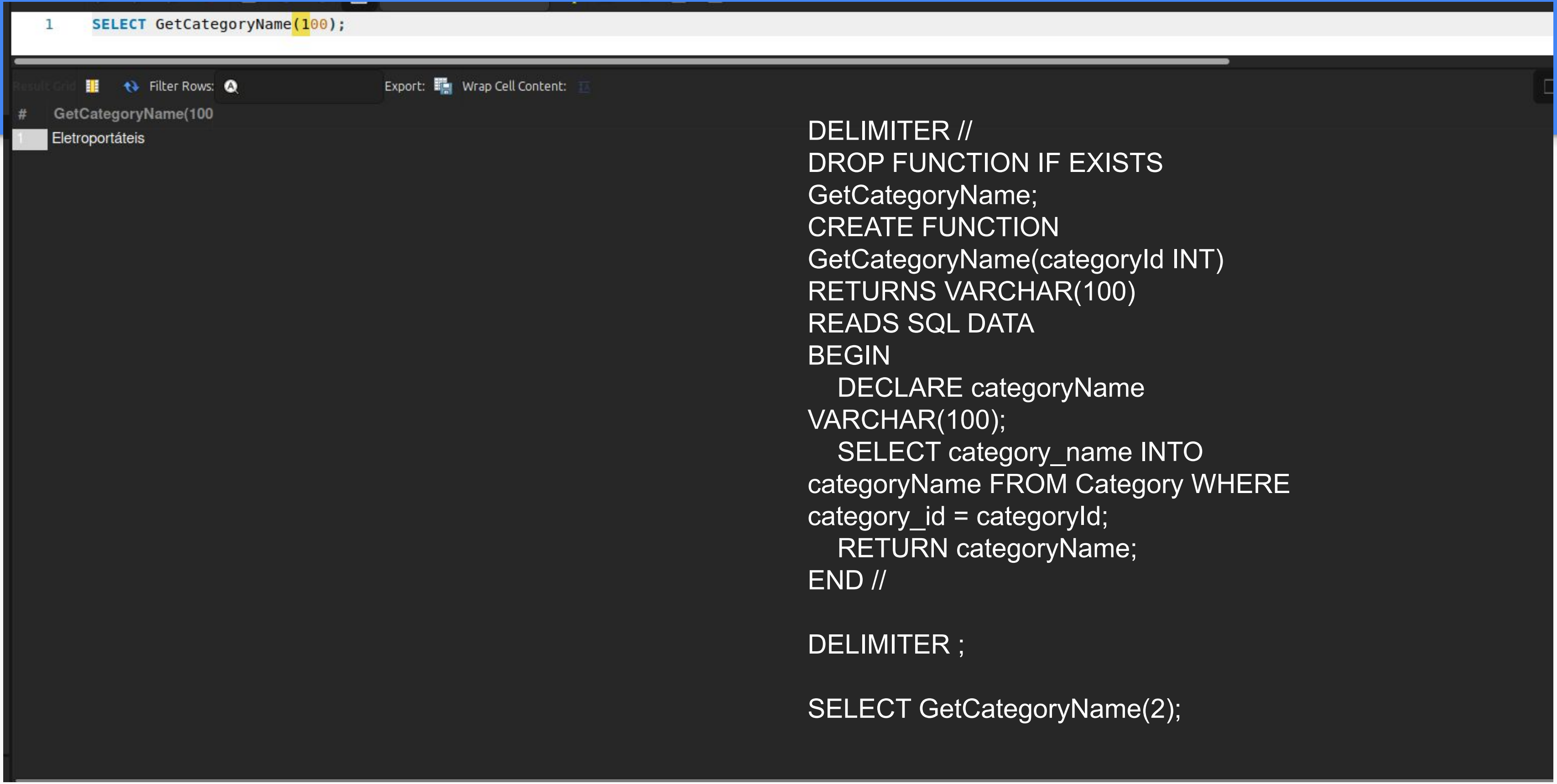


```
SELECT CategoriaComMaisProdutos();
```

```
#   CategoriaComMaisProdutos
1   1
```

```
DELIMITER //

CREATE FUNCTION
CategoriaComMaisProdutos()
RETURNS INT
READS SQL DATA
BEGIN
    DECLARE categoria_mais_popular INT;
    SELECT category_id INTO
categoria_mais_popular
    FROM Product
    GROUP BY category_id
    ORDER BY COUNT(*) DESC
    LIMIT 1;
    RETURN categoria_mais_popular;
END //

DELIMITER ;
```

Functions - GetCategoryName(2)

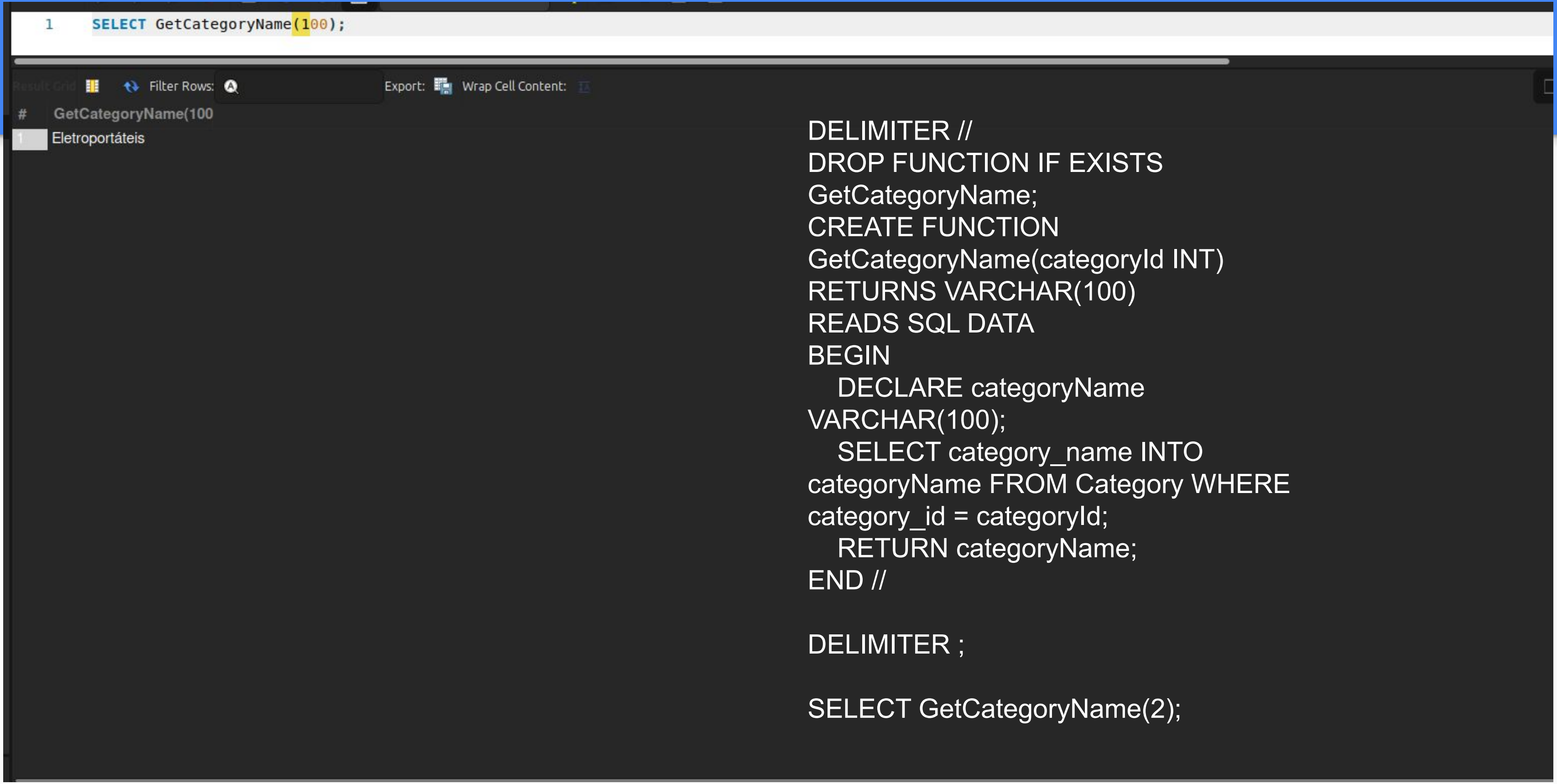


```
SELECT GetCategoryName(100);
```

```
#  GetCategoryName(100
1  Eletroportáteis
```

```
DELIMITER //
DROP FUNCTION IF EXISTS
GetCategoryName;
CREATE FUNCTION
GetCategoryName(categoryId INT)
RETURNS VARCHAR(100)
READS SQL DATA
BEGIN
    DECLARE categoryName
VARCHAR(100);
    SELECT category_name INTO
categoryName FROM Category WHERE
category_id = categoryId;
    RETURN categoryName;
END //

DELIMITER ;

SELECT GetCategoryName(2);
```

Functions - GetCategoryName(2)

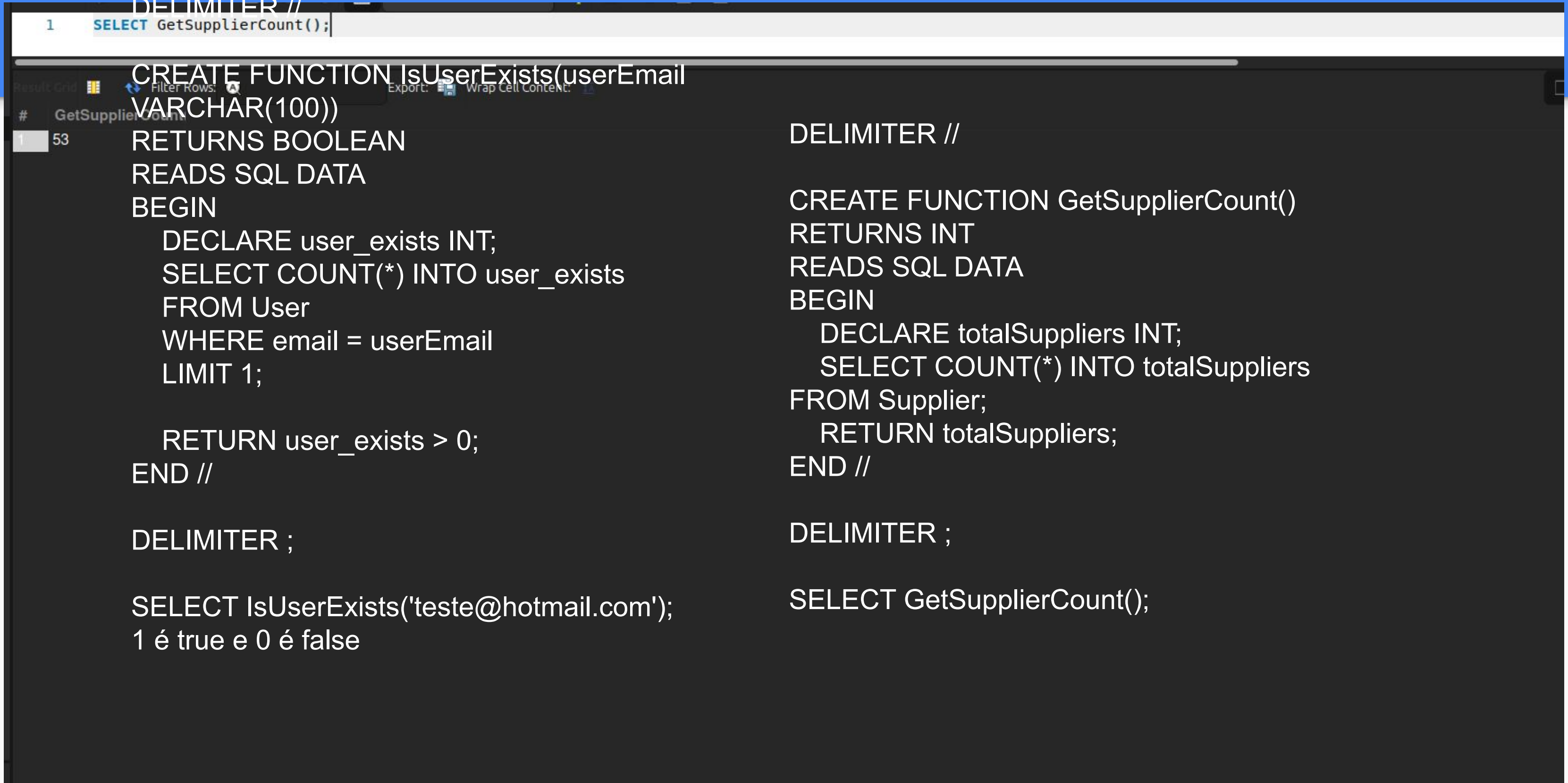


```
SELECT GetCategoryName(100);
```

```
#   GetCategoryName(100
1   Eletroportáteis
```

```
DELIMITER //
DROP FUNCTION IF EXISTS
GetCategoryName;
CREATE FUNCTION
GetCategoryName(categoryId INT)
RETURNS VARCHAR(100)
READS SQL DATA
BEGIN
    DECLARE categoryName
VARCHAR(100);
    SELECT category_name INTO
categoryName FROM Category WHERE
category_id = categoryId;
    RETURN categoryName;
END //

DELIMITER ;

SELECT GetCategoryName(2);
```

Functions - GetSupplierCount()



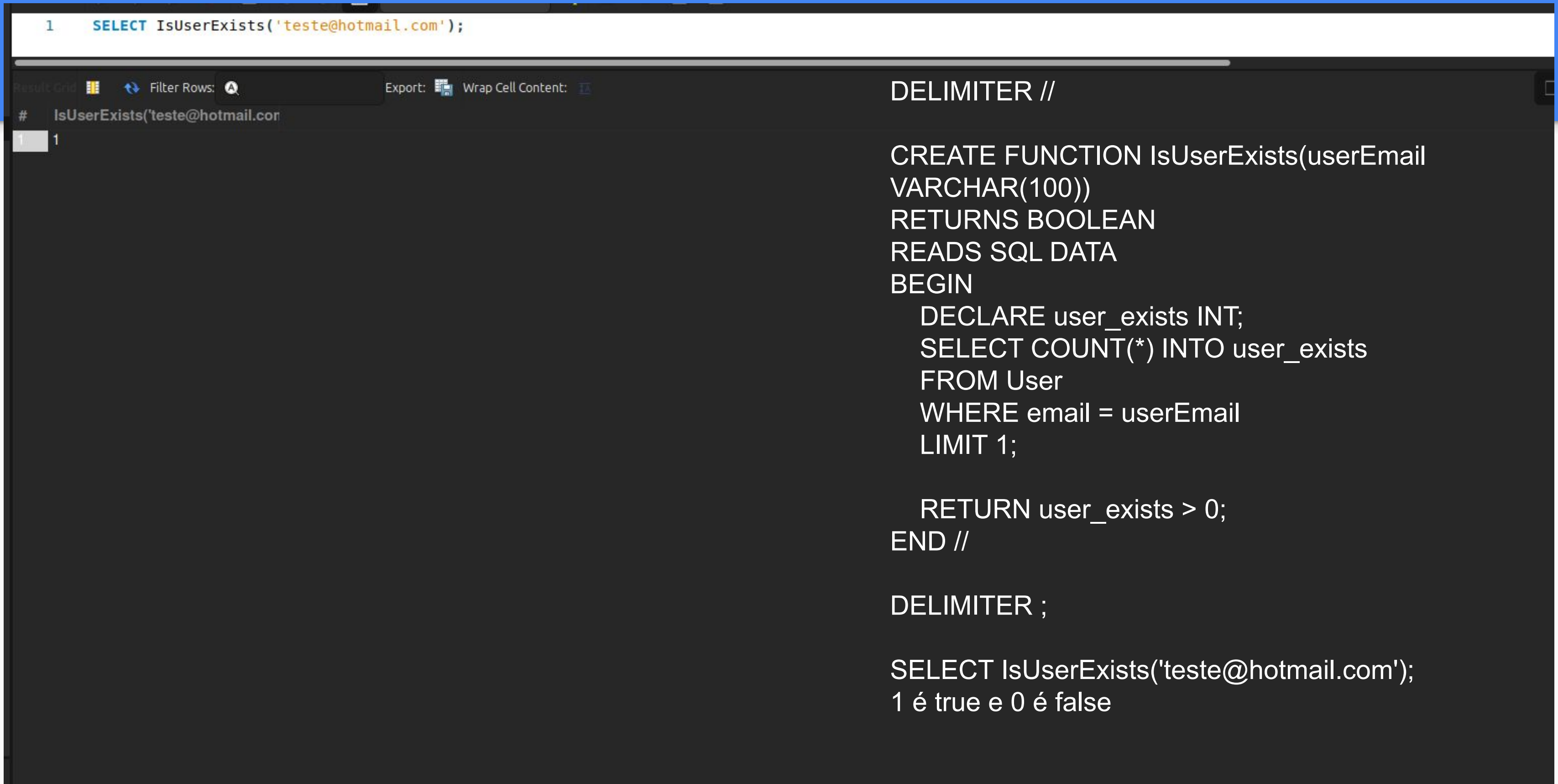
```
SELECT GetSupplierCount();
```

#	GetSupplierCount
1	53

```
DELIMITER //

CREATE FUNCTION IsUserExists(userEmail
VARCHAR(100))
RETURNS BOOLEAN
READS SQL DATA
BEGIN
    DECLARE user_exists INT;
    SELECT COUNT(*) INTO user_exists
    FROM User
    WHERE email = userEmail
    LIMIT 1;

    RETURN user_exists > 0;
END //

DELIMITER ;

SELECT IsUserExists('teste@hotmail.com');
1 é true e 0 é false
```

```
DELIMITER //

CREATE FUNCTION GetSupplierCount()
RETURNS INT
READS SQL DATA
BEGIN
    DECLARE totalSuppliers INT;
    SELECT COUNT(*) INTO totalSuppliers
FROM Supplier;
    RETURN totalSuppliers;
END //

DELIMITER ;

SELECT GetSupplierCount();
```

Functions - IsUserExists()



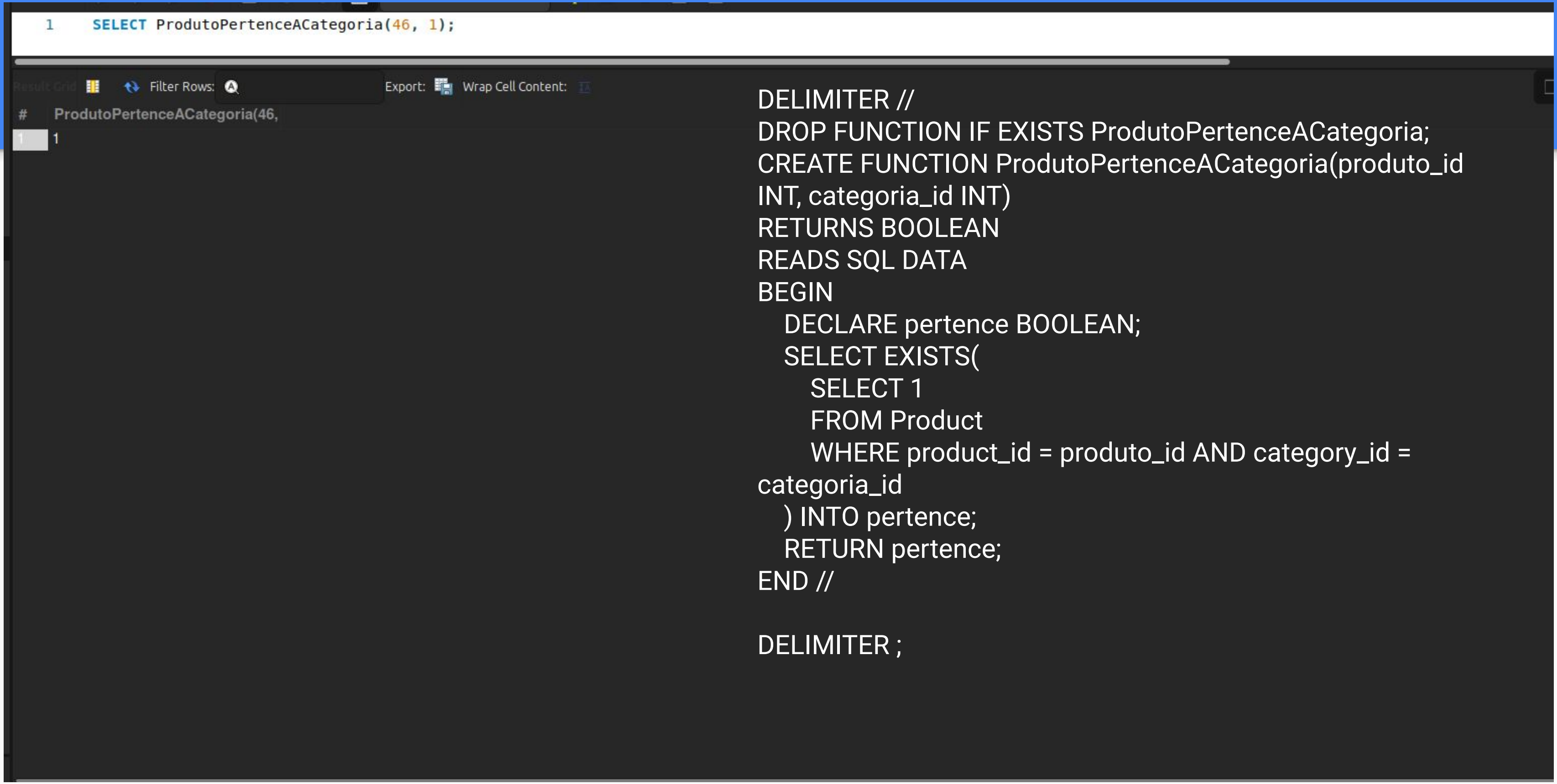
```
SELECT IsUserExists('teste@hotmail.com');
```

```
# IsUserExists('teste@hotmail.con
1 1
```

```
DELIMITER //

CREATE FUNCTION IsUserExists(userEmail
VARCHAR(100))
RETURNS BOOLEAN
READS SQL DATA
BEGIN
    DECLARE user_exists INT;
    SELECT COUNT(*) INTO user_exists
    FROM User
    WHERE email = userEmail
    LIMIT 1;

    RETURN user_exists > 0;
END //

DELIMITER ;

SELECT IsUserExists('teste@hotmail.com');
```

1 é true e 0 é false

Functions - ProdutoPertenceACategoria()

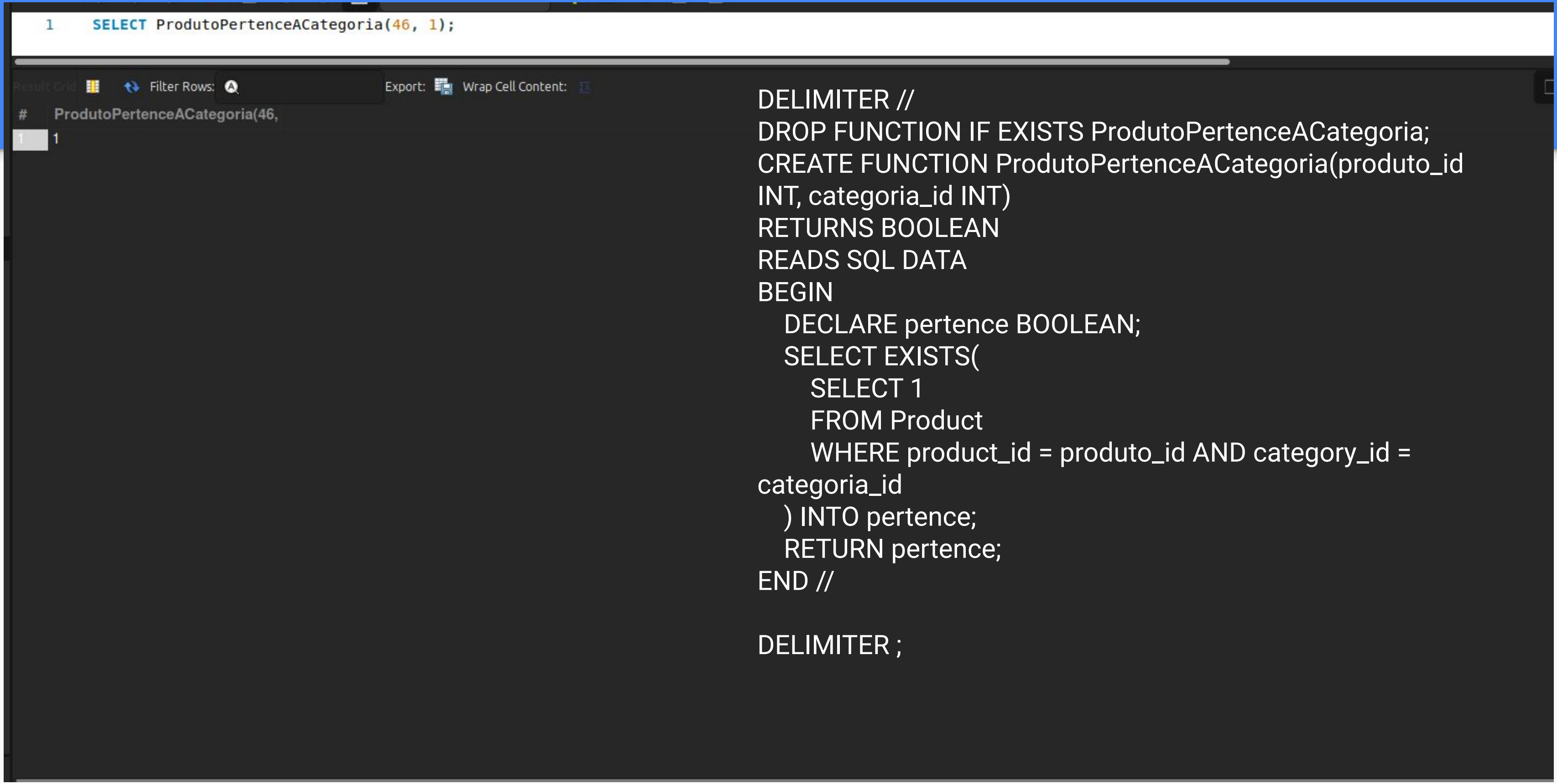


```sql
SELECT ProdutoPertenceACategoria(46, 1);
```

```
#   ProdutoPertenceACategoria(46,
1   1
```

```sql
DELIMITER //
DROP FUNCTION IF EXISTS ProdutoPertenceACategoria;
CREATE FUNCTION ProdutoPertenceACategoria(produto_id
INT, categoria_id INT)
RETURNS BOOLEAN
READS SQL DATA
BEGIN
    DECLARE pertence BOOLEAN;
    SELECT EXISTS(
        SELECT 1
        FROM Product
        WHERE product_id = produto_id AND category_id =
categoria_id
    ) INTO pertence;
    RETURN pertence;
END //

DELIMITER ;
```

Functions - ProdutoPertenceACategoria()

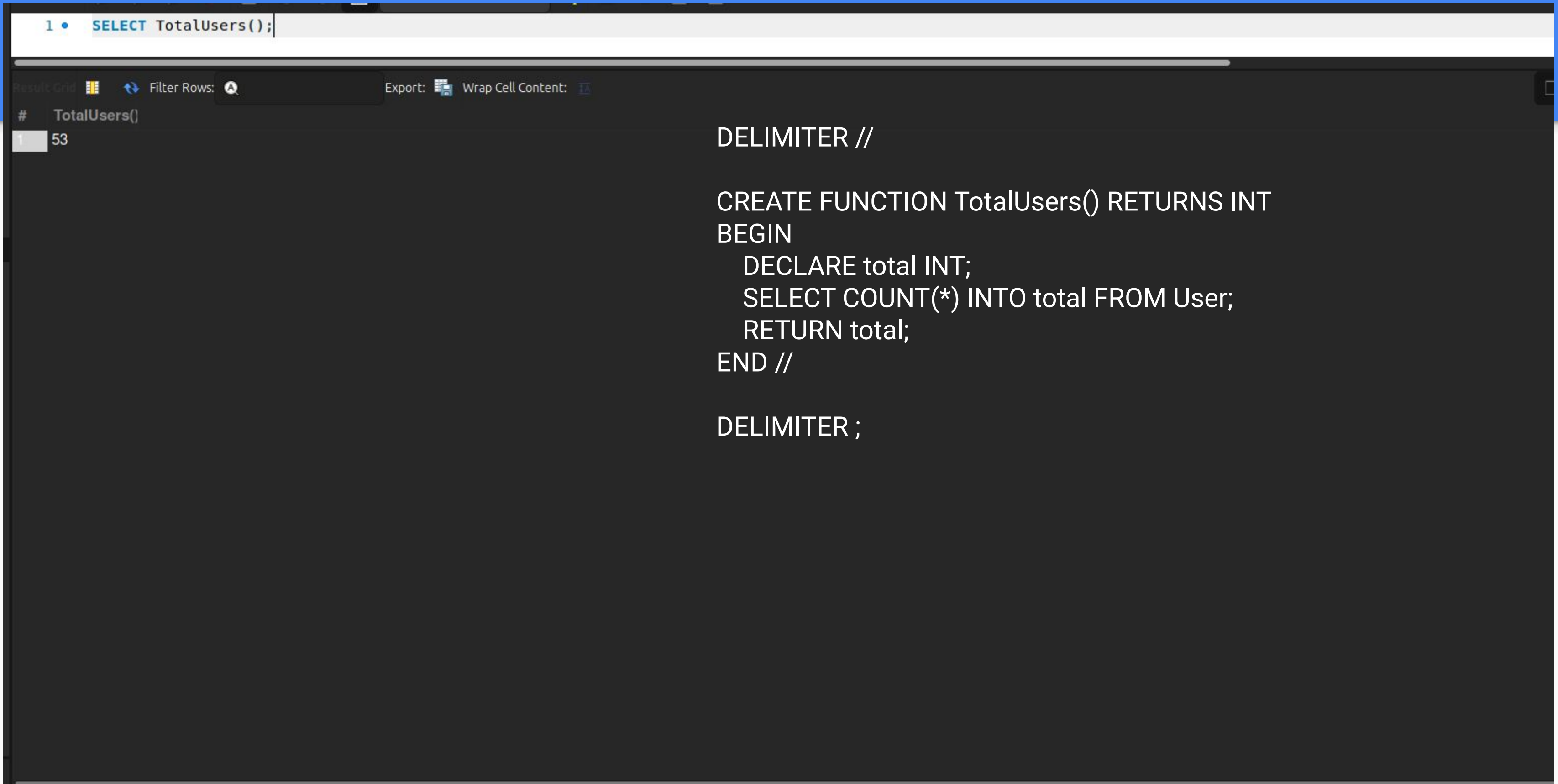


```
SELECT ProdutoPertenceACategoria(46, 1);
```

```
#   ProdutoPertenceACategoria(46,
1   1
```

```
DELIMITER //
DROP FUNCTION IF EXISTS ProdutoPertenceACategoria;
CREATE FUNCTION ProdutoPertenceACategoria(produto_id
INT, categoria_id INT)
RETURNS BOOLEAN
READS SQL DATA
BEGIN
    DECLARE pertence BOOLEAN;
    SELECT EXISTS(
        SELECT 1
        FROM Product
        WHERE product_id = produto_id AND category_id =
categoria_id
    ) INTO pertence;
    RETURN pertence;
END //

DELIMITER ;
```

Functions -TotalUsers()

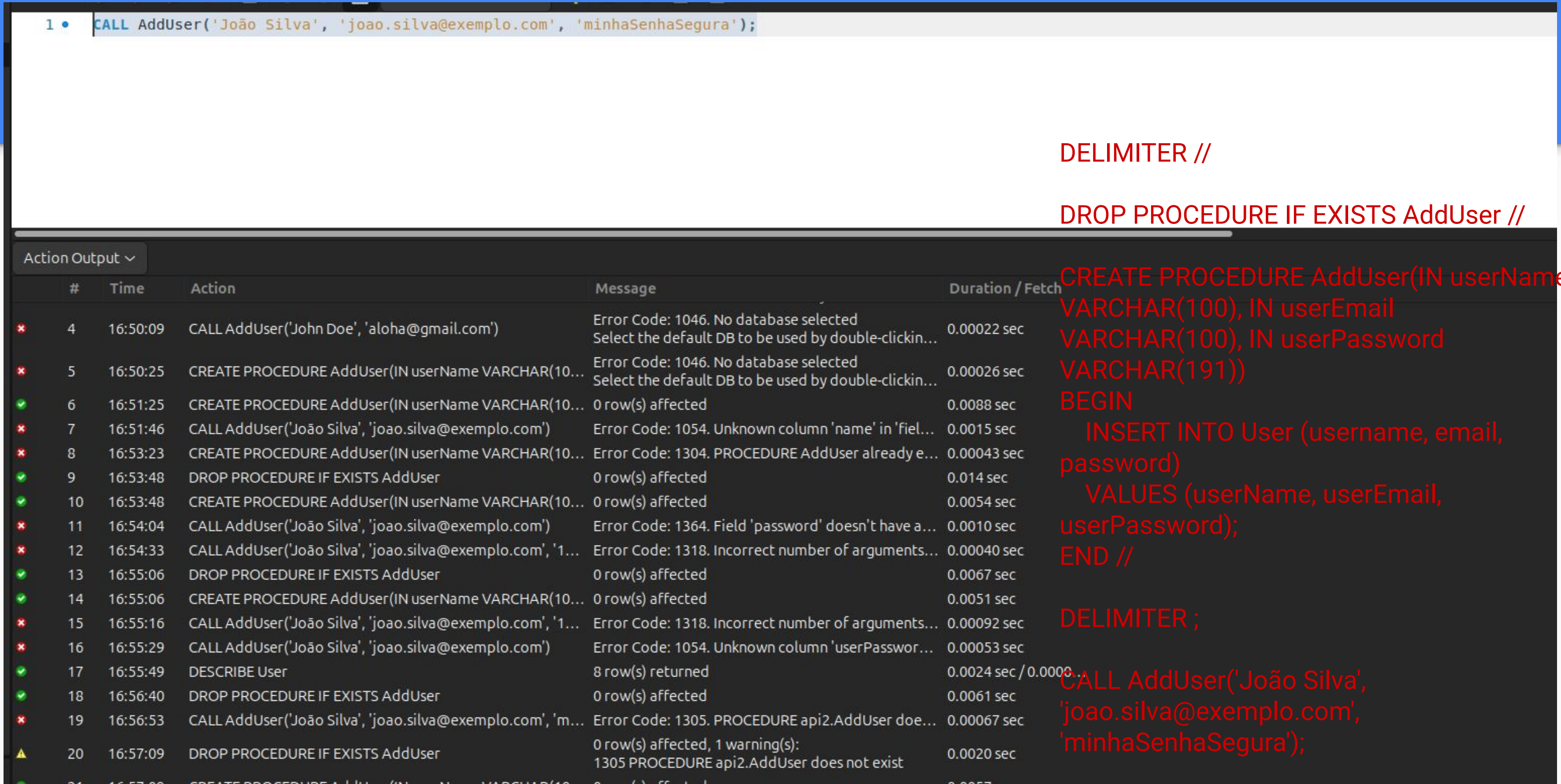


```
SELECT TotalUsers();
```

```
#   TotalUsers()
1   53
```

```
DELIMITER //

CREATE FUNCTION TotalUsers() RETURNS INT
BEGIN
    DECLARE total INT;
    SELECT COUNT(*) INTO total FROM User;
    RETURN total;
END //

DELIMITER ;
```

Procedures - AddUser

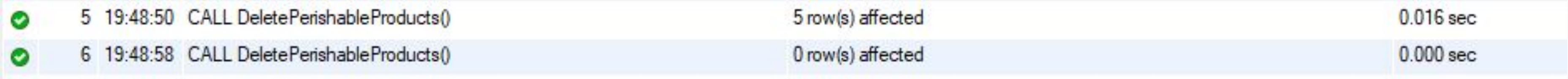


```
CALL AddUser('João Silva', 'joao.silva@exemplo.com', 'minhaSenhaSegura');
```

	#	Time	Action	Message	Duration / Fetch
✗	4	16:50:09	CALL AddUser('John Doe', 'aloha@gmail.com')	Error Code: 1046. No database selected Select the default DB to be used by double-clickin...	0.00022 sec
✗	5	16:50:25	CREATE PROCEDURE AddUser(IN userName VARCHAR(10...	Error Code: 1046. No database selected Select the default DB to be used by double-clickin...	0.00026 sec
✓	6	16:51:25	CREATE PROCEDURE AddUser(IN userName VARCHAR(10...	0 row(s) affected	0.0088 sec
✗	7	16:51:46	CALL AddUser('João Silva', 'joao.silva@exemplo.com')	Error Code: 1054. Unknown column 'name' in 'fiel...	0.0015 sec
✗	8	16:53:23	CREATE PROCEDURE AddUser(IN userName VARCHAR(10...	Error Code: 1304. PROCEDURE AddUser already e...	0.00043 sec
✓	9	16:53:48	DROP PROCEDURE IF EXISTS AddUser	0 row(s) affected	0.014 sec
✓	10	16:53:48	CREATE PROCEDURE AddUser(IN userName VARCHAR(10...	0 row(s) affected	0.0054 sec
✗	11	16:54:04	CALL AddUser('João Silva', 'joao.silva@exemplo.com')	Error Code: 1364. Field 'password' doesn't have a...	0.0010 sec
✗	12	16:54:33	CALL AddUser('João Silva', 'joao.silva@exemplo.com', '1...	Error Code: 1318. Incorrect number of arguments...	0.00040 sec
✓	13	16:55:06	DROP PROCEDURE IF EXISTS AddUser	0 row(s) affected	0.0067 sec
✓	14	16:55:06	CREATE PROCEDURE AddUser(IN userName VARCHAR(10...	0 row(s) affected	0.0051 sec
✗	15	16:55:16	CALL AddUser('João Silva', 'joao.silva@exemplo.com', '1...	Error Code: 1318. Incorrect number of arguments...	0.00092 sec
✗	16	16:55:29	CALL AddUser('João Silva', 'joao.silva@exemplo.com')	Error Code: 1054. Unknown column 'userPasswor...	0.00053 sec
✓	17	16:55:49	DESCRIBE User	8 row(s) returned	0.0024 sec / 0.0000...
✓	18	16:56:40	DROP PROCEDURE IF EXISTS AddUser	0 row(s) affected	0.0061 sec
✗	19	16:56:53	CALL AddUser('João Silva', 'joao.silva@exemplo.com', 'm...	Error Code: 1305. PROCEDURE api2.AddUser doe...	0.00067 sec
⚠	20	16:57:09	DROP PROCEDURE IF EXISTS AddUser	0 row(s) affected, 1 warning(s): 1305 PROCEDURE api2.AddUser does not exist	0.0020 sec

```
DELIMITER //

DROP PROCEDURE IF EXISTS AddUser //

CREATE PROCEDURE AddUser(IN userName VARCHAR(100), IN userEmail VARCHAR(100), IN userPassword VARCHAR(191))
BEGIN
    INSERT INTO User (username, email, password)
    VALUES (userName, userEmail, userPassword);
END //

DELIMITER ;

CALL AddUser('João Silva', 'joao.silva@exemplo.com', 'minhaSenhaSegura');
```

Procedures - DeletePerishableProducts

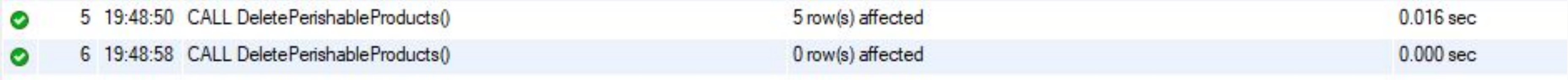


```
✓  5  19:48:50  CALL DeletePerishableProducts()    5 row(s) affected    0.016 sec
✓  6  19:48:58  CALL DeletePerishableProducts()    0 row(s) affected    0.000 sec
```

```
INSERT INTO Product (product_name, description, product_img,
category_id, prod_brand, prod_model, supplier_id, unit_id, is_perishable,
prod_cost_value, prod_sell_value)
VALUES
('Fresh Milk', '1L Whole Milk', 'milk.jpg', 5, 'Dairy Farm', 'Fresh Milk', 1,
FLOOR(1 + RAND() * 3), 1, 1.50, 2.25),
('Organic Yogurt', 'Greek Yogurt, Low-Fat', 'yogurt.jpg', 5, 'Healthy Foods',
'Organic Yogurt', 1, FLOOR(1 + RAND() * 3), 1, 2.00, 3.50),
('Fresh Berries', 'Mixed Berries, 1lb', 'berries.jpg', 5, 'Local Farm', 'Fresh
Berries', 1, FLOOR(1 + RAND() * 3), 1, 3.00, 4.50),
('Fresh Fish', 'Salmon Fillet, 1lb', 'fish.jpg', 5, 'Ocean Harvest', 'Fresh
Salmon', 1, FLOOR(1 + RAND() * 3), 1, 5.00, 8.00),
('Fresh Vegetables', 'Mixed Vegetables, 1lb', 'veggies.jpg', 5, 'Green
Grocer', 'Fresh Veggies', 1, FLOOR(1 + RAND() * 3), 1, 2.50, 4.00);
```

```
DELIMITER //

CREATE PROCEDURE
GetProductsByCategory(IN
categoryId INT)
BEGIN
    SELECT * FROM Product
WHERE category_id = categoryId;
END //

DELIMITER ;
```

Procedures - DeleteProductByCategory



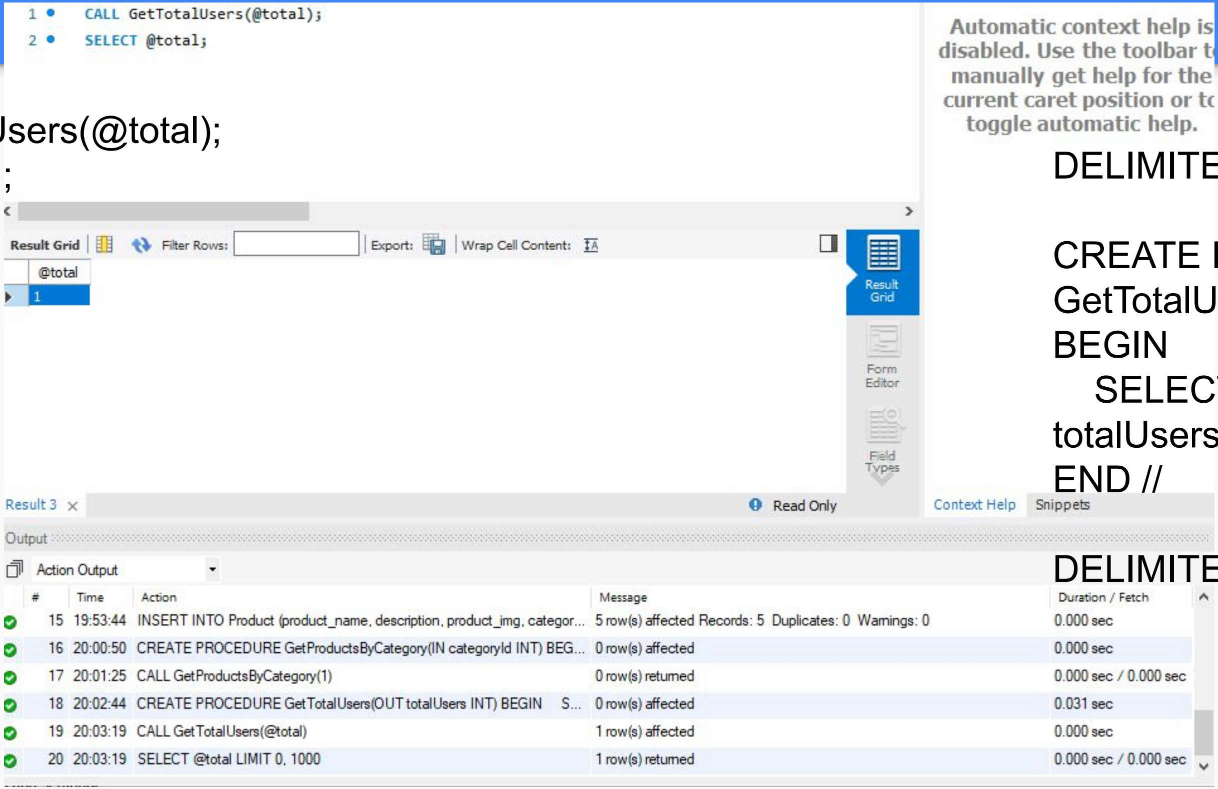
```
✓  5  19:48:50  CALL DeletePerishableProducts()     5 row(s) affected     0.016 sec
✓  6  19:48:58  CALL DeletePerishableProducts()     0 row(s) affected     0.000 sec
```

```
INSERT INTO Product (product_name, description, product_img,
category_id, prod_brand, prod_model, supplier_id, unit_id, is_perishable,
prod_cost_value, prod_sell_value)
VALUES
('Fresh Milk', '1L Whole Milk', 'milk.jpg', 5, 'Dairy Farm', 'Fresh Milk', 1,
FLOOR(1 + RAND() * 3), 1, 1.50, 2.25),
('Organic Yogurt', 'Greek Yogurt, Low-Fat', 'yogurt.jpg', 5, 'Healthy Foods',
'Organic Yogurt', 1, FLOOR(1 + RAND() * 3), 1, 2.00, 3.50),
('Fresh Berries', 'Mixed Berries, 1lb', 'berries.jpg', 5, 'Local Farm', 'Fresh
Berries', 1, FLOOR(1 + RAND() * 3), 1, 3.00, 4.50),
('Fresh Fish', 'Salmon Fillet, 1lb', 'fish.jpg', 5, 'Ocean Harvest', 'Fresh
Salmon', 1, FLOOR(1 + RAND() * 3), 1, 5.00, 8.00),
('Fresh Vegetables', 'Mixed Vegetables, 1lb', 'veggies.jpg', 5, 'Green
Grocer', 'Fresh Veggies', 1, FLOOR(1 + RAND() * 3), 1, 2.50, 4.00);
```

```
DELIMITER //

CREATE PROCEDURE
GetProductsByCategory(IN
categoryId INT)
BEGIN
    SELECT * FROM Product
WHERE category_id = categoryId;
END //

DELIMITER ;
```

Procedures - gettotalusers

CALL GetTotalUsers(@total);
SELECT @total;



```
CALL GetTotalUsers(@total);
SELECT @total;
```

Automatic context help is disabled. Use the toolbar t manually get help for the current caret position or tc toggle automatic help.

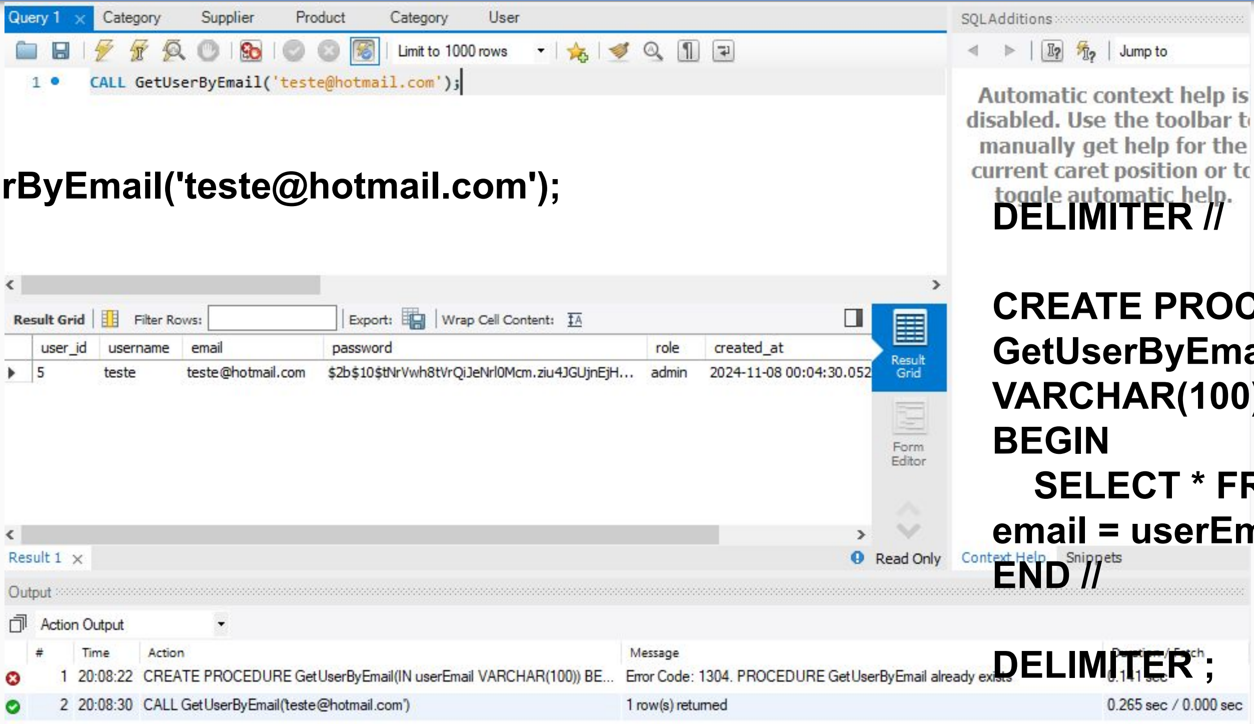
@total
1

#	Time	Action	Message	Duration / Fetch
15	19:53:44	INSERT INTO Product (product_name, description, product_img, categor...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.000 sec
16	20:00:50	CREATE PROCEDURE GetProductsByCategory(IN categoryId INT) BEG...	0 row(s) affected	0.000 sec
17	20:01:25	CALL GetProductsByCategory(1)	0 row(s) returned	0.000 sec / 0.000 sec
18	20:02:44	CREATE PROCEDURE GetTotalUsers(OUT totalUsers INT) BEGIN S...	0 row(s) affected	0.031 sec
19	20:03:19	CALL GetTotalUsers(@total)	1 row(s) affected	0.000 sec
20	20:03:19	SELECT @total LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

```
DELIMITER //

CREATE PROCEDURE
GetTotalUsers(OUT totalUsers INT)
BEGIN
    SELECT COUNT(*) INTO
totalUsers FROM User;
END //

DELIMITER ;
```

Procedures - getuserbyemail

CALL GetUserByEmail('teste@hotmail.com');



```
CALL GetUserByEmail('teste@hotmail.com');
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

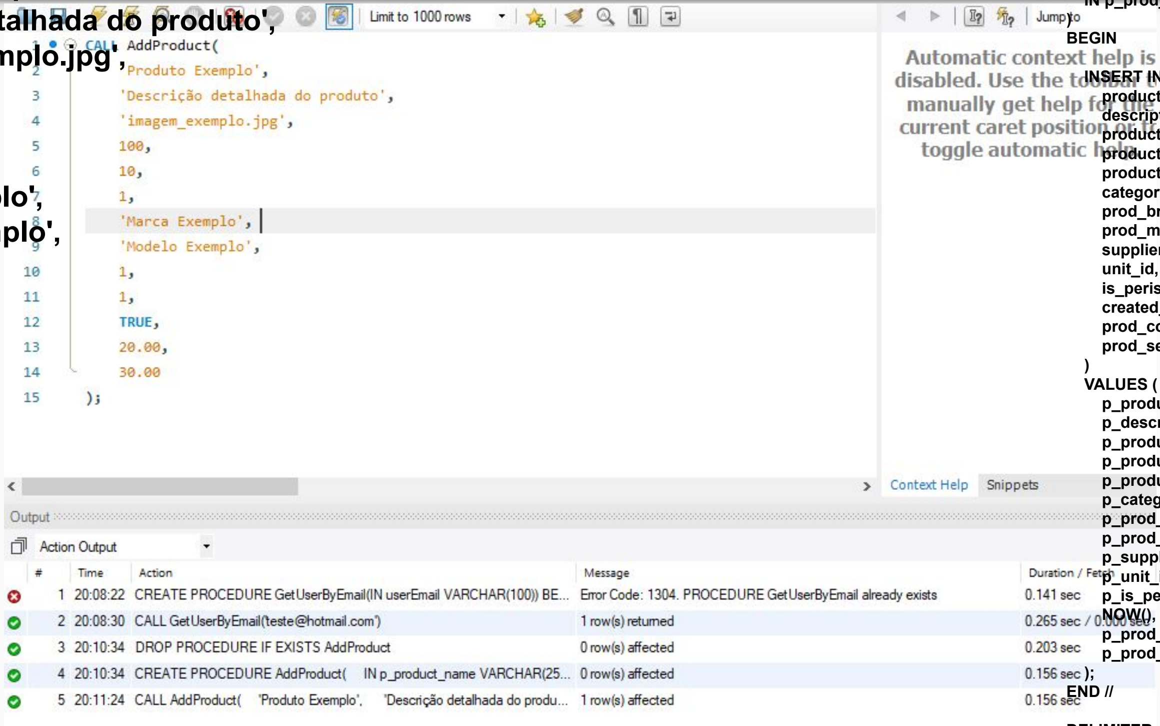
	user_id	username	email	password	role	created_at
▶	5	teste	teste@hotmail.com	$2b$10$tNrVwh8tVrQiJeNrl0Mcm.ziu4JGUjnEjH...	admin	2024-11-08 00:04:30.052

	#	Time	Action	Message	Duration / Fetch
☒	1	20:08:22	CREATE PROCEDURE GetUserByEmail(IN userEmail VARCHAR(100)) BE...	Error Code: 1304. PROCEDURE GetUserByEmail already exists	0.141 sec
✓	2	20:08:30	CALL GetUserByEmail('teste@hotmail.com')	1 row(s) returned	0.265 sec / 0.000 sec

```
DELIMITER //

CREATE PROCEDURE GetUserByEmail(IN userEmail VARCHAR(100))
BEGIN
    SELECT * FROM User WHERE email = userEmail;
END //

DELIMITER ;
```

Procedures - updateproductstock

```
CALL AddProduct(
    'Produto Exemplo',
    'Descrição detalhada do produto',
    'imagem_exemplo.jpg',
    100,
    10,
    1,
    'Marca Exemplo',
    'Modelo Exemplo',
    1,
    1,
    TRUE,
    20.00,
    30.00
);
```



```
CALL AddProduct(
    'Produto Exemplo',
    'Descrição detalhada do produto',
    'imagem_exemplo.jpg',
    100,
    10,
    1,
    'Marca Exemplo',
    'Modelo Exemplo',
    1,
    1,
    TRUE,
    20.00,
    30.00
);
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```
    #  Time      Action                                                        Message                                                     Duration / Fetch
    1  20:08:22  CREATE PROCEDURE GetUserByEmail(IN userEmail VARCHAR(100)) BE...  Error Code: 1304. PROCEDURE GetUserByEmail already exists    0.141 sec
    2  20:08:30  CALL GetUserByEmail('teste@hotmail.com')                       1 row(s) returned                                           0.265 sec / 0.000 sec
    3  20:10:34  DROP PROCEDURE IF EXISTS AddProduct                            0 row(s) affected                                           0.203 sec
    4  20:10:34  CREATE PROCEDURE AddProduct(    IN p_product_name VARCHAR(25...  0 row(s) affected                                          0.156 sec
    5  20:11:24  CALL AddProduct(    'Produto Exemplo',    'Descrição detalhada do produ...  1 row(s) affected                             0.156 sec
```

```
DROP PROCEDURE IF EXISTS AddProduct;

DELIMITER //

CREATE PROCEDURE AddProduct(
  IN p_product_name VARCHAR(255),
  IN p_description TEXT,
  IN p_product_img VARCHAR(255),
  IN p_product_stock INT,
  IN p_product_stock_min INT,
  IN p_category_id INT,
  IN p_prod_brand VARCHAR(255),
  IN p_prod_model VARCHAR(255),
  IN p_supplier_id INT,
  IN p_unit_id INT,
  IN p_is_perishable BOOLEAN,
  IN p_prod_cost_value DECIMAL(10, 2),
  IN p_prod_sell_value DECIMAL(10, 2)
)
BEGIN
  INSERT INTO Product (
    product_name,
    description,
    product_img,
    product_stock,
    product_stock_min,
    category_id,
    prod_brand,
    prod_model,
    supplier_id,
    unit_id,
    is_perishable,
    created_at,
    prod_cost_value,
    prod_sell_value
  )
  VALUES (
    p_product_name,
    p_description,
    p_product_img,
    p_product_stock,
    p_product_stock_min,
    p_category_id,
    p_prod_brand,
    p_prod_model,
    p_supplier_id,
    p_unit_id,
    p_is_perishable,
    NOW(),
    p_prod_cost_value,
    p_prod_sell_value
  );
END //

DELIMITER
```

Triggers - AfterDelectProducts

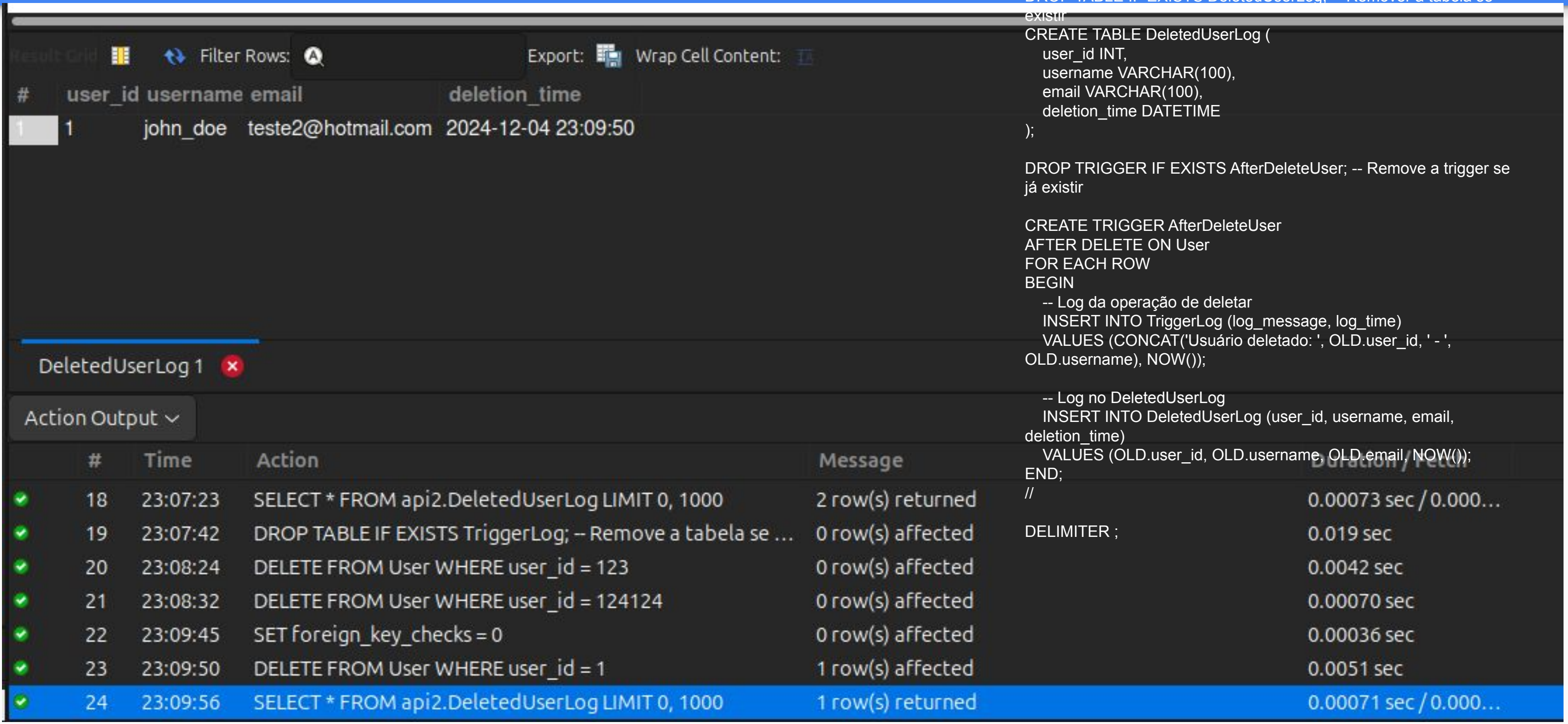
```
DROP TRIGGER IF EXISTS
AfterDeleteProduct;

-- Criação da trigger para registrar a exclusão
de produtos
DELIMITER //

CREATE TRIGGER AfterDeleteProduct
AFTER DELETE ON Product
FOR EACH ROW
BEGIN
   -- Inserir registro na tabela Trigger ao
deletar um produto
         INSERT INTO TriggerLog
(log_message, log_time)
         VALUES (CONCAT('Produto com ID ',
OLD.product_id, ' e nome ''',
OLD.product_name, ''' foi deletado.'),
NOW());
END //
```

```
UPDATE Batch SET product_id =
NULL WHERE product_id = 50;
DELETE FROM Product WHERE
product_id = 50;
```

Triggers - AfterDelectUser

```
SET foreign_key_checks = 0;
DELETE FROM User WHERE user_id = 123;
```

```
DELIMITER //

DROP TABLE IF EXISTS TriggerLog; -- Remove a tabela se ela já
existir

CREATE TABLE TriggerLog (
    log_id INT AUTO_INCREMENT PRIMARY KEY,
    log_message VARCHAR(255),
    log_time DATETIME
);

-- Se você ainda não tiver a tabela DeletedUserLog, crie-a aqui
DROP TABLE IF EXISTS DeletedUserLog; -- Remover a tabela se
existir
CREATE TABLE DeletedUserLog (
    user_id INT,
    username VARCHAR(100),
    email VARCHAR(100),
    deletion_time DATETIME
);

DROP TRIGGER IF EXISTS AfterDeleteUser; -- Remove a trigger se
já existir

CREATE TRIGGER AfterDeleteUser
AFTER DELETE ON User
FOR EACH ROW
BEGIN
    -- Log da operação de deletar
    INSERT INTO TriggerLog (log_message, log_time)
    VALUES (CONCAT('Usuário deletado: ', OLD.user_id, ' - ',
OLD.username), NOW());

    -- Log no DeletedUserLog
    INSERT INTO DeletedUserLog (user_id, username, email,
deletion_time)
    VALUES (OLD.user_id, OLD.username, OLD.email, NOW());
END;
//

DELIMITER ;
```

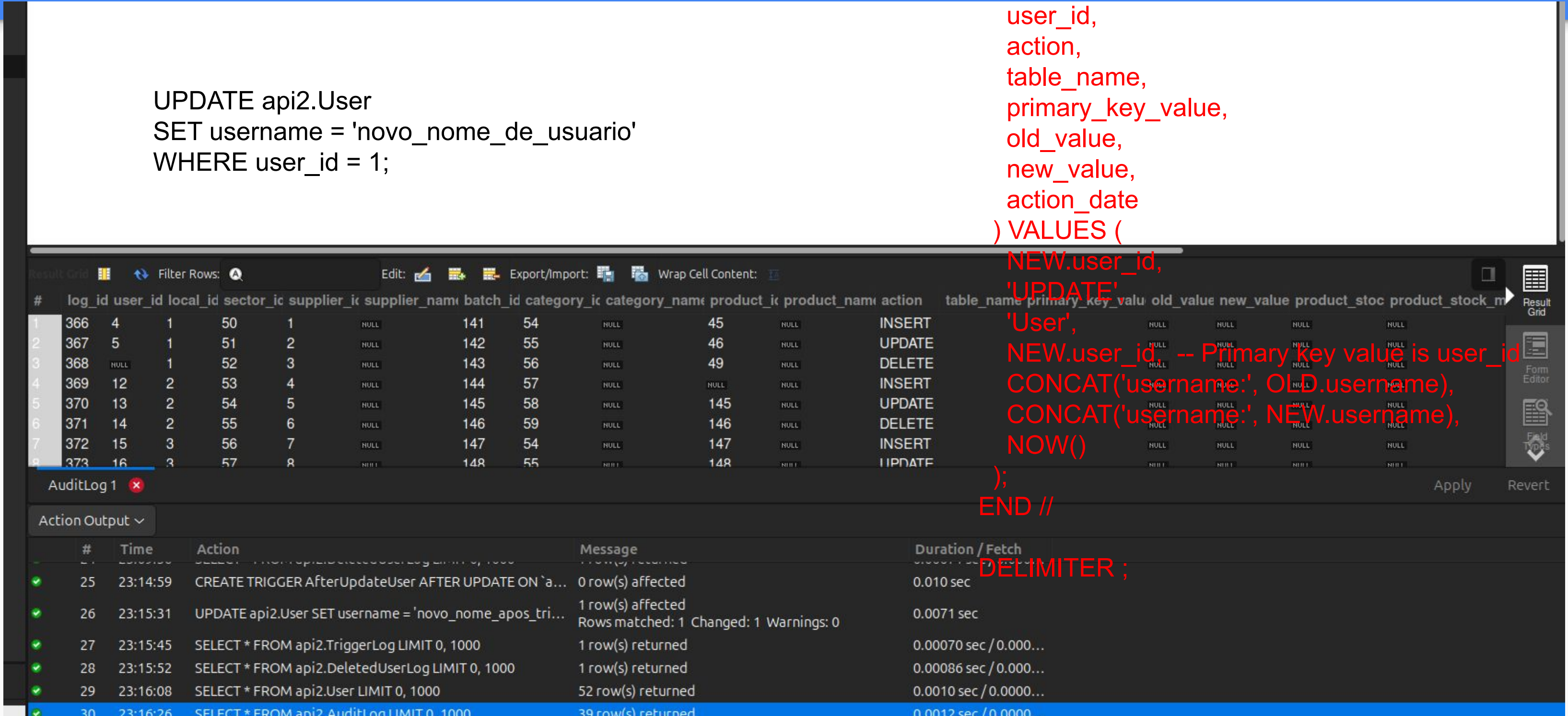


#	user_id	username	email	deletion_time
1	1	john_doe	teste2@hotmail.com	2024-12-04 23:09:50

#	Time	Action	Message	Duration / Fetch
18	23:07:23	SELECT * FROM api2.DeletedUserLog LIMIT 0, 1000	2 row(s) returned	0.00073 sec / 0.000...
19	23:07:42	DROP TABLE IF EXISTS TriggerLog; -- Remove a tabela se ...	0 row(s) affected	0.019 sec
20	23:08:24	DELETE FROM User WHERE user_id = 123	0 row(s) affected	0.0042 sec
21	23:08:32	DELETE FROM User WHERE user_id = 124124	0 row(s) affected	0.00070 sec
22	23:09:45	SET foreign_key_checks = 0	0 row(s) affected	0.00036 sec
23	23:09:50	DELETE FROM User WHERE user_id = 1	1 row(s) affected	0.0051 sec
24	23:09:56	SELECT * FROM api2.DeletedUserLog LIMIT 0, 1000	1 row(s) returned	0.00071 sec / 0.000...

Triggers - AfterUpdateUser

```
DELIMITER //

CREATE TRIGGER AfterUpdateUser
AFTER UPDATE ON `api2`.`User`
FOR EACH ROW
BEGIN
  -- Insert into AuditLog table instead of UserLog
(assuming it exists)
  INSERT INTO `api2`.`AuditLog` (
    user_id,
    action,
    table_name,
    primary_key_value,
    old_value,
    new_value,
    action_date
  ) VALUES (
    NEW.user_id,
    'UPDATE',
    'User',
    NEW.user_id,  -- Primary key value is user_id
    CONCAT('username:', OLD.username),
    CONCAT('username:', NEW.username),
    NOW()
  );
END //

DELIMITER ;
```



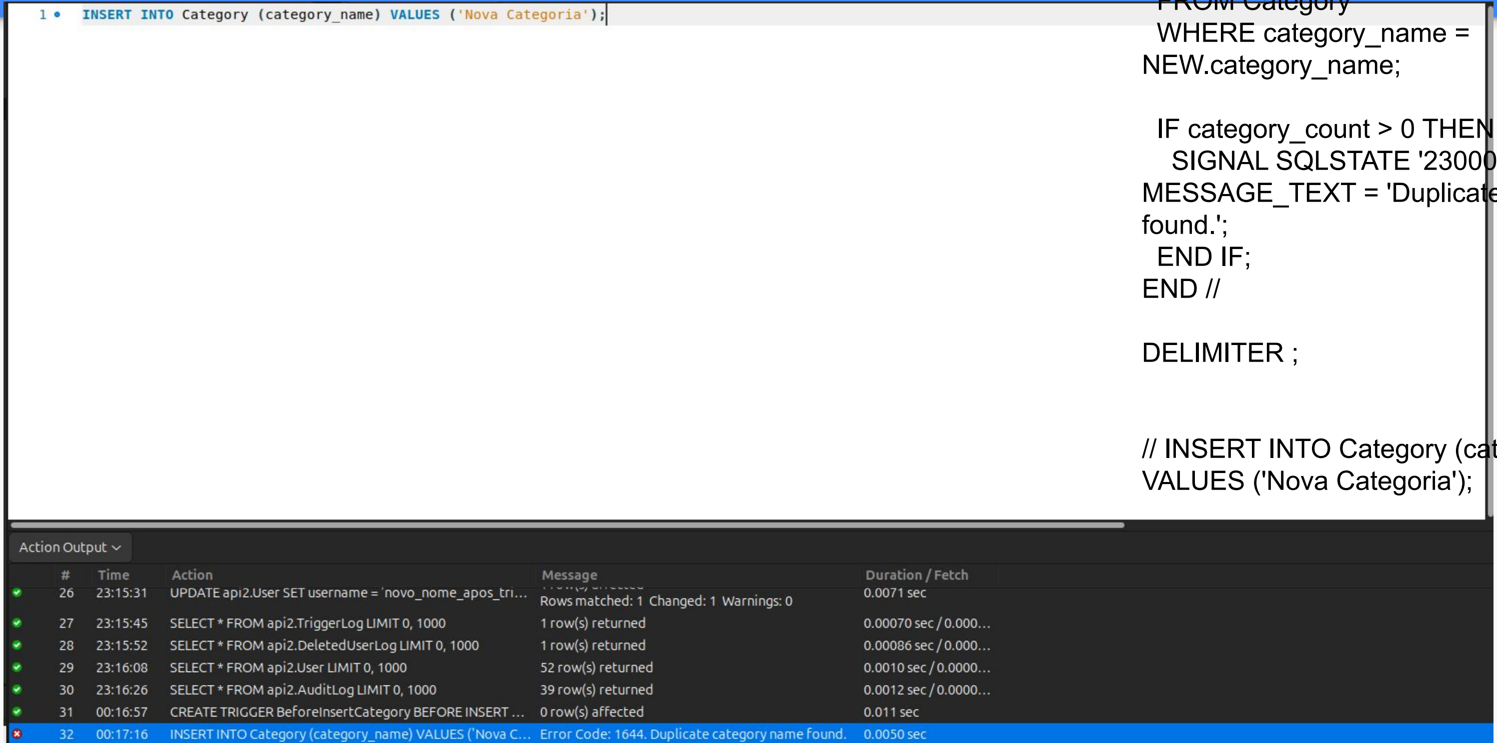
```
UPDATE api2.User
SET username = 'novo_nome_de_usuario'
WHERE user_id = 1;
```

#	log_id	user_id	local_id	sector_ic	supplier_ic	supplier_nam	batch_id	category_ic	category_name	product_ic	product_nam	action	table_name	primary_key_valu	old_value	new_value	product_stoc	product_stock_m
1	366	4	1	50	1	NULL	141	54	NULL	45	NULL	INSERT	[illegible]	[illegible]	NULL	NULL	NULL	NULL
2	367	5	1	51	2	NULL	142	55	NULL	46	NULL	UPDATE	[illegible]	[illegible]	NULL	NULL	NULL	NULL
3	368	NULL	1	52	3	NULL	143	56	NULL	49	NULL	DELETE	[illegible]	[illegible]	NULL	NULL	NULL	NULL
4	369	12	2	53	4	NULL	144	57	NULL	NULL	NULL	INSERT	[illegible]	[illegible]	NULL	NULL	NULL	NULL
5	370	13	2	54	5	NULL	145	58	NULL	145	NULL	UPDATE	[illegible]	[illegible]	NULL	NULL	NULL	NULL
6	371	14	2	55	6	NULL	146	59	NULL	146	NULL	DELETE	[illegible]	[illegible]	NULL	NULL	NULL	NULL
7	372	15	3	56	7	NULL	147	54	NULL	147	NULL	INSERT	[illegible]	[illegible]	NULL	NULL	NULL	NULL
8	373	16	3	57	8	NULL	148	55	NULL	148	NULL	UPDATE	[illegible]	[illegible]	NULL	NULL	NULL	NULL

```
#   Time      Action                                              Message                                          Duration / Fetch
25  23:14:59  CREATE TRIGGER AfterUpdateUser AFTER UPDATE ON `a...  0 row(s) affected                                0.010 sec
26  23:15:31  UPDATE api2.User SET username = 'novo_nome_apos_tri...  1 row(s) affected                             0.0071 sec
                                                                  Rows matched: 1  Changed: 1  Warnings: 0
27  23:15:45  SELECT * FROM api2.TriggerLog LIMIT 0, 1000         1 row(s) returned                                0.00070 sec / 0.000...
28  23:15:52  SELECT * FROM api2.DeletedUserLog LIMIT 0, 1000     1 row(s) returned                                0.00086 sec / 0.000...
29  23:16:08  SELECT * FROM api2.User LIMIT 0, 1000               52 row(s) returned                               0.0010 sec / 0.0000...
30  23:16:26  SELECT * FROM api2.AuditLog LIMIT 0, 1000           39 row(s) returned                               0.0012 sec / 0.0000
```

Triggers - BeforeInsertCategory

```
DELIMITER //

CREATE TRIGGER BeforeInsertCategory
BEFORE INSERT ON Category
FOR EACH ROW
BEGIN
  DECLARE category_count INT;

  SELECT COUNT(*) INTO category_count
  FROM Category
  WHERE category_name =
NEW.category_name;

  IF category_count > 0 THEN
    SIGNAL SQLSTATE '23000' SET
MESSAGE_TEXT = 'Duplicate category name
found.';
  END IF;
END //

DELIMITER ;


// INSERT INTO Category (category_name)
VALUES ('Nova Categoria');
```

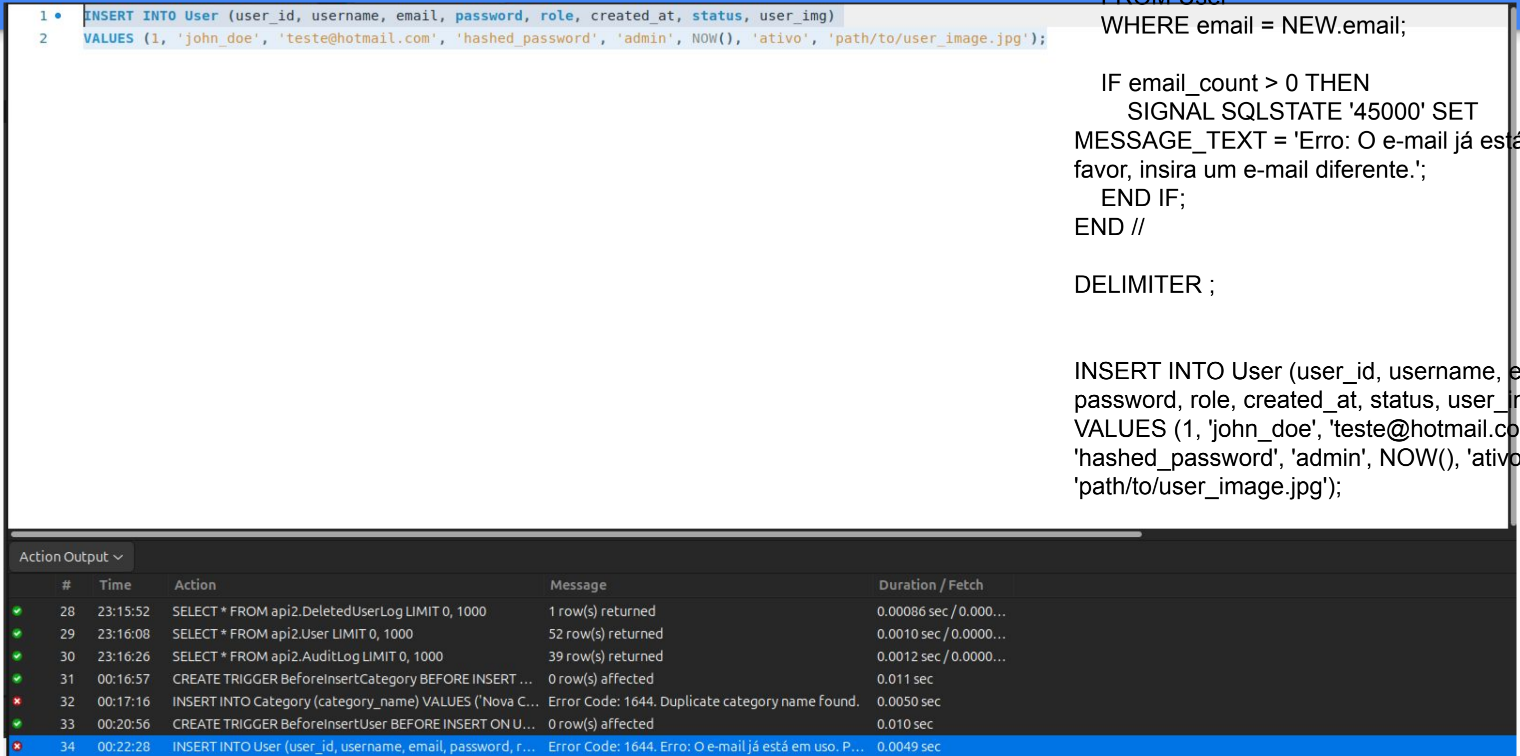


```
INSERT INTO Category (category_name) VALUES ('Nova Categoria');
```

```
    #   Time      Action                                                  Message                                         Duration / Fetch
    26  23:15:31  UPDATE api2.User SET username = 'novo_nome_apos_tri...   Rows matched: 1  Changed: 1  Warnings: 0        0.0071 sec
    27  23:15:45  SELECT * FROM api2.TriggerLog LIMIT 0, 1000             1 row(s) returned                               0.00070 sec / 0.000...
    28  23:15:52  SELECT * FROM api2.DeletedUserLog LIMIT 0, 1000         1 row(s) returned                               0.00086 sec / 0.000...
    29  23:16:08  SELECT * FROM api2.User LIMIT 0, 1000                   52 row(s) returned                              0.0010 sec / 0.0000...
    30  23:16:26  SELECT * FROM api2.AuditLog LIMIT 0, 1000               39 row(s) returned                              0.0012 sec / 0.0000...
    31  00:16:57  CREATE TRIGGER BeforeInsertCategory BEFORE INSERT ...   0 row(s) affected                               0.011 sec
    32  00:17:16  INSERT INTO Category (category_name) VALUES ('Nova C...  Error Code: 1644. Duplicate category name found.  0.0050 sec
```

Triggers - BeforeInsertUser

```
DELIMITER //

CREATE TRIGGER BeforeInsertUser
BEFORE INSERT ON User
FOR EACH ROW
BEGIN
   DECLARE email_count INT;

   SELECT COUNT(*)
   INTO email_count
   FROM User
   WHERE email = NEW.email;

   IF email_count > 0 THEN
      SIGNAL SQLSTATE '45000' SET
MESSAGE_TEXT = 'Erro: O e-mail já está em uso. Por
favor, insira um e-mail diferente.';
   END IF;
END //

DELIMITER ;


INSERT INTO User (user_id, username, email,
password, role, created_at, status, user_img)
VALUES (1, 'john_doe', 'teste@hotmail.com',
'hashed_password', 'admin', NOW(), 'ativo',
'path/to/user_image.jpg');
```

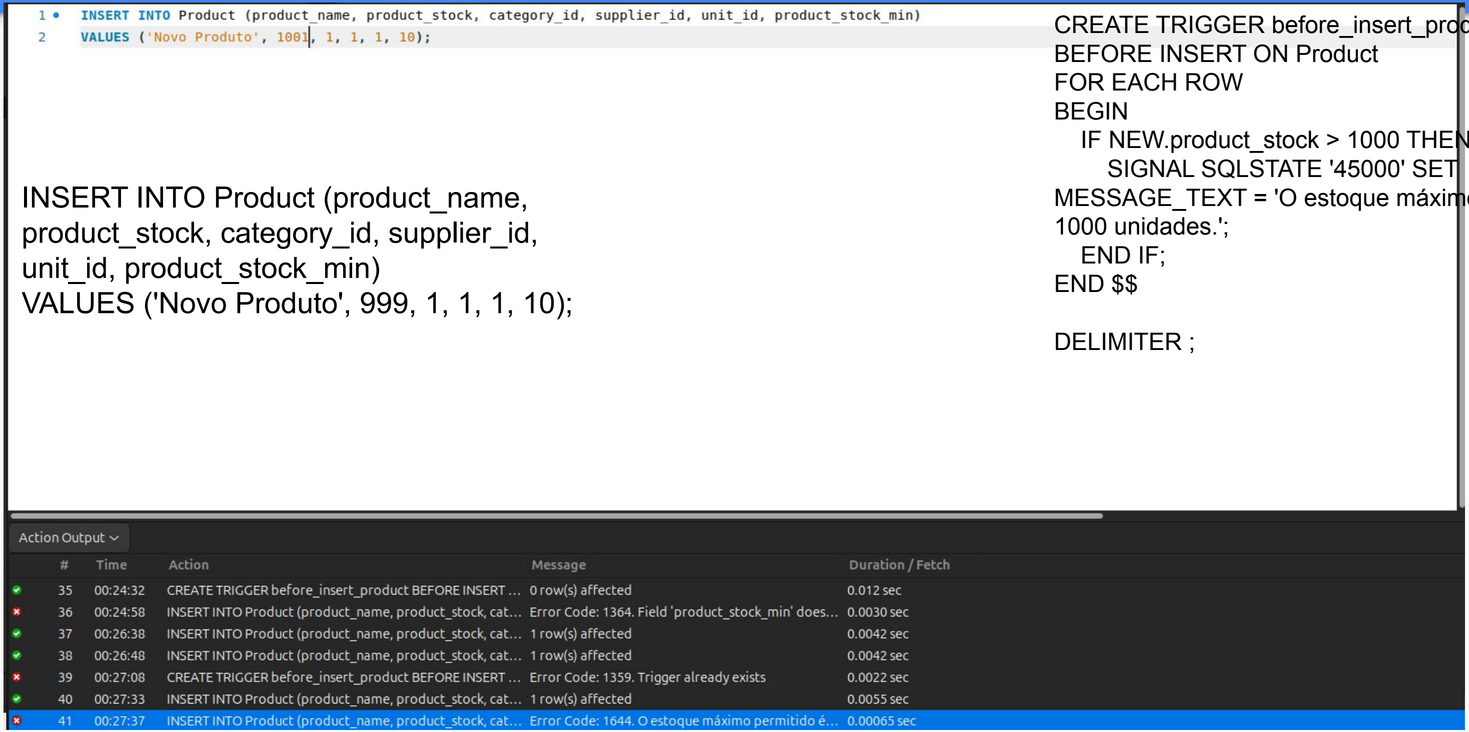


```
INSERT INTO User (user_id, username, email, password, role, created_at, status, user_img)
VALUES (1, 'john_doe', 'teste@hotmail.com', 'hashed_password', 'admin', NOW(), 'ativo', 'path/to/user_image.jpg');
```

Action Output

```
   #   Time      Action                                             Message                                                Duration / Fetch
✓  28  23:15:52  SELECT * FROM api2.DeletedUserLog LIMIT 0, 1000    1 row(s) returned                                      0.00086 sec / 0.000...
✓  29  23:16:08  SELECT * FROM api2.User LIMIT 0, 1000              52 row(s) returned                                     0.0010 sec / 0.0000...
✓  30  23:16:26  SELECT * FROM api2.AuditLog LIMIT 0, 1000          39 row(s) returned                                     0.0012 sec / 0.0000...
✓  31  00:16:57  CREATE TRIGGER BeforeInsertCategory BEFORE INSERT ...  0 row(s) affected                                 0.011 sec
✗  32  00:17:16  INSERT INTO Category (category_name) VALUES ('Nova C...  Error Code: 1644. Duplicate category name found.  0.0050 sec
✓  33  00:20:56  CREATE TRIGGER BeforeInsertUser BEFORE INSERT ON U...  0 row(s) affected                                  0.010 sec
✗  34  00:22:28  INSERT INTO User (user_id, username, email, password, r...  Error Code: 1644. Erro: O e-mail já está em uso. P...  0.0049 sec
```

Triggers - BeforeInsertProduct



```
INSERT INTO Product (product_name, product_stock, category_id, supplier_id, unit_id, product_stock_min)
VALUES ('Novo Produto', 1001, 1, 1, 1, 10);
```

Action Output

	#	Time	Action	Message	Duration / Fetch
✓	35	00:24:32	CREATE TRIGGER before_insert_product BEFORE INSERT …	0 row(s) affected	0.012 sec
✗	36	00:24:58	INSERT INTO Product (product_name, product_stock, cat…	Error Code: 1364. Field 'product_stock_min' does…	0.0030 sec
✓	37	00:26:38	INSERT INTO Product (product_name, product_stock, cat…	1 row(s) affected	0.0042 sec
✓	38	00:26:48	INSERT INTO Product (product_name, product_stock, cat…	1 row(s) affected	0.0042 sec
✗	39	00:27:08	CREATE TRIGGER before_insert_product BEFORE INSERT …	Error Code: 1359. Trigger already exists	0.0022 sec
✓	40	00:27:33	INSERT INTO Product (product_name, product_stock, cat…	1 row(s) affected	0.0055 sec
✗	41	00:27:37	INSERT INTO Product (product_name, product_stock, cat…	Error Code: 1644. O estoque máximo permitido é…	0.00065 sec

```
INSERT INTO Product (product_name,
product_stock, category_id, supplier_id,
unit_id, product_stock_min)
VALUES ('Novo Produto', 999, 1, 1, 1, 10);
```

```
DELIMITER $$

CREATE TRIGGER before_insert_product
BEFORE INSERT ON Product
FOR EACH ROW
BEGIN
   IF NEW.product_stock > 1000 THEN
      SIGNAL SQLSTATE '45000' SET
MESSAGE_TEXT = 'O estoque máximo permitido é
1000 unidades.';
   END IF;
END $$

DELIMITER ;
```