



# Provisões de Emergência

Kelvin Lehrback



# Principais pontos

1. Contribuir com o planejamento financeiro de famílias e ONGS.
2. Verificar quais os produtos essenciais desse grupo de pessoas (dessas famílias).
3. Relatórios de preços dos produtos.
4. Relatórios de fornecedores dos produtos.
5. Relatório dos produtos mais requisitados.
6. Verificar o total de gastos de cada família.

# Prototipação e Tabela de dados

**MINHA CASA**  
**RELATÓRIO**

Intervalo de tempo  
Mensal

Produtos	Total R\$
Alimentícios	300,75
Medicinais	150,00
Limpeza	146,46
<b>Total</b>	<b>597,21</b>

KLG

**MINHA CASA**  
**PESSOAS**

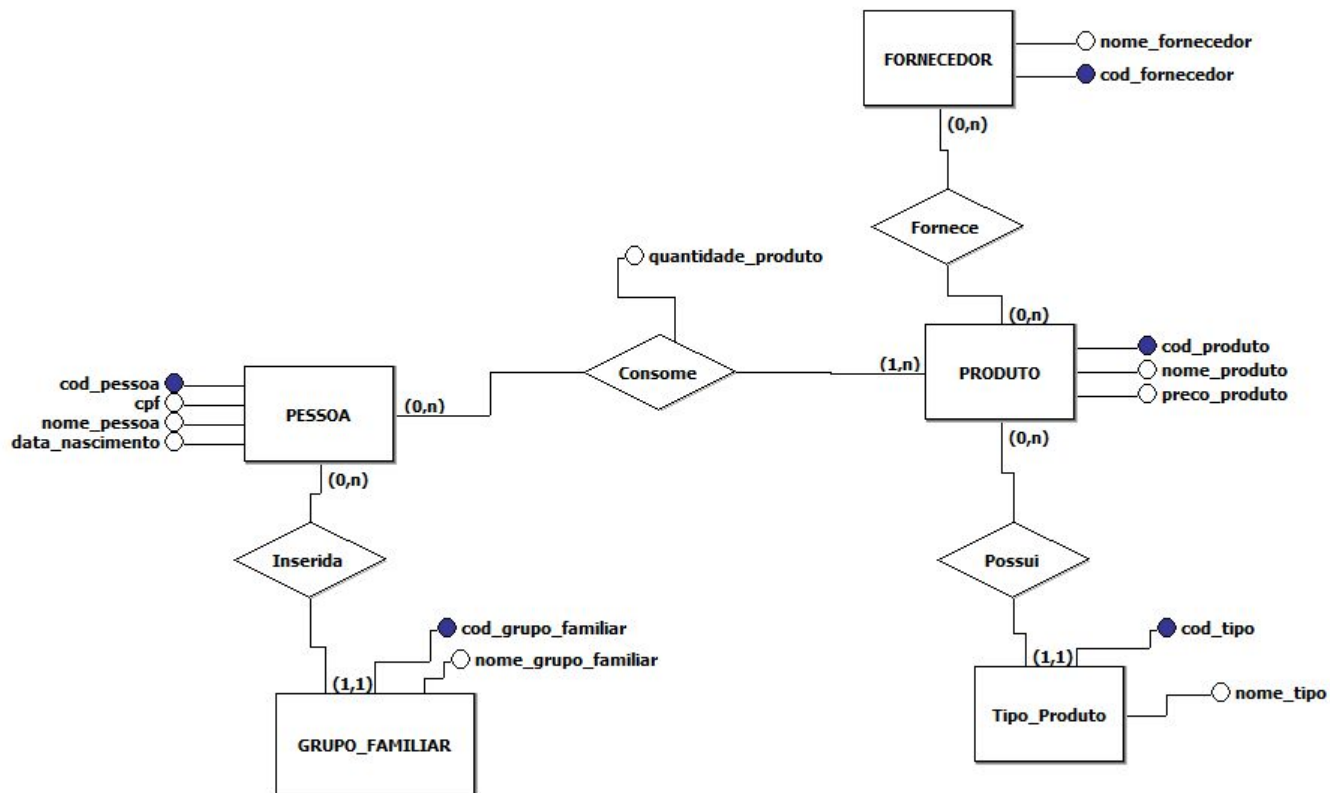
Nome	Idade
Joel Silva	43
Ellie Johson	15
Marta Silva	38
Apollo Johson	64

+

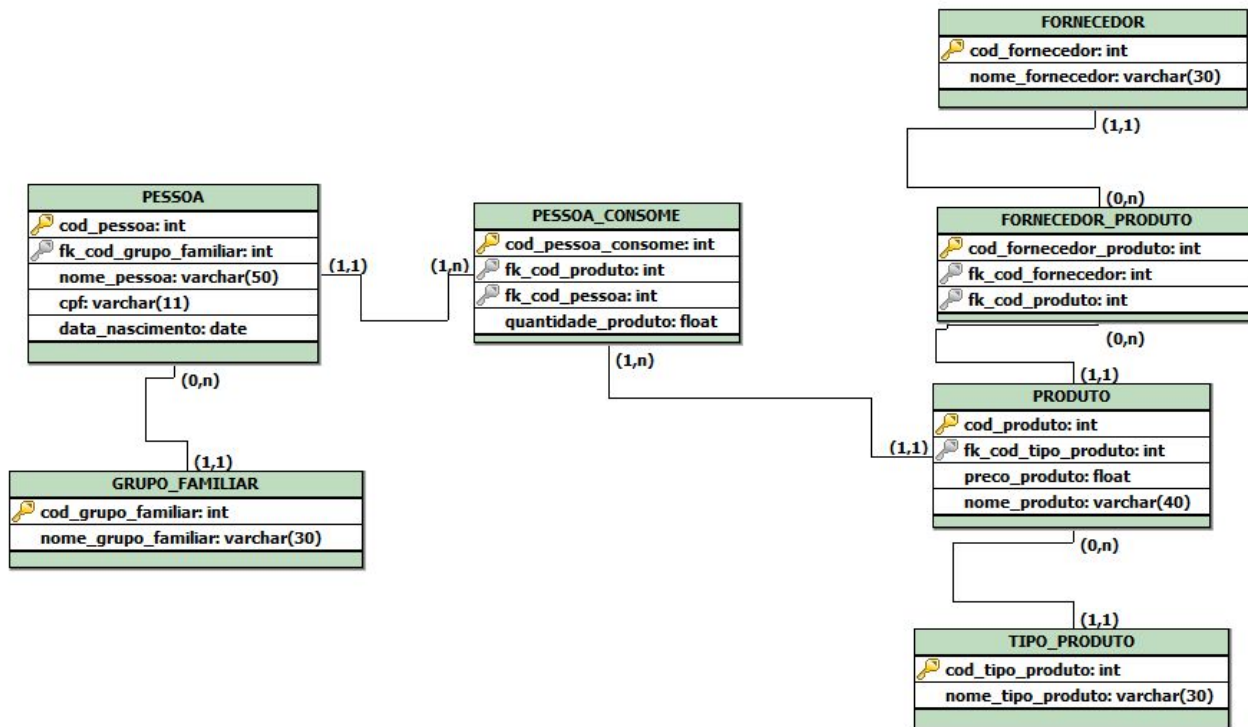
KLG

A	B	C	D	E	F	G	H	I	J
PESSOA					TIPO_PRODUTO		FORNECEDOR_PRODUTO		
cod_pessoa	fk_cod_grupo_familiar	nome_pessoa	cpf	data_nascimento	cod_tipo_produto	nome_tipo_produto	cod_fornecedor_p roduto	fk_cod_fornecedor	fk_cod_produ to
1	1	Kelvin L	83087427853	15/08/1998	1	Alimentício	1	1	1
2	1	Valdeci G	61576870880	16/08/1998	2	Medicinal	2	1	2
3	1	João C	80663532787	17/08/1998	3	Limpeza geral	3	2	5
4	1	Clara L	6410440979	18/08/1998	4	Limpeza Pessoal	4	3	3
5	2	Sebastian da Silva	69505791607	12/02/1986	GRUPO_FAMILIAR		5	3	6
6	2	Eliot S	14101087830	28/01/1988	cod_grupo_familiar	nome_grupo_familiar	6	4	4
7	2	Tião Moreira	92251924876	02/05/1983	1	Minha casa	7	5	10
8	2	Haley Quinn	2407861736	05/06/1996	2	Familia de Pescadores - Barra de VV	8	6	7
9	2	Nair P	27148449286	29/05/1993	PESSOA_CONSOME				
10	2	Fabio Brazza	10176554130	08/09/1998	cod_pessoa_conso me	fk_cod_pessoa	fk_cod_produto	quantidade_produto	
PRODUTO					1	1	1	300g	
cod_produto	fk_cod_tipo_produto	nome_produto	preco_produto		2	2	2	200g	
1	1	Arroz	7.79		3	3	4	1 dose	
2	1	Feijão	5.99		4	4	3	1 unidade	
3	4	Pasta de dente	3.79		5	3	5	100g	
4	2	Antibiótico	17.03		6	2	6	250ml	
5	1	Peixe	37.90		7	4	5	100g	
6	3	Desinfetante	2.59		8	1	2	200g	
7	1	Macarrão	4.39		9	2	1	200g	
8	1	Farinha	5.39		10	5	7	200g	
10	1	Maça	2.38		11	6	8	100g	
11	2	Insulina	6.99		12	7	9	8 unidades	
12	4	Alcool em gel	11.99		13	8	10	3ml	
13	3	Cloro	10.79		14	9	11	100 ml	
FORNECEDOR					15	10	12	500ml	
cod_fornecedor	nome_fornecedor				16	5	1	200g	
1	MegaG				17	6	2	300g	
2	Ryu Alimentos				18	7	3	1 unidades	
3	OTC				19	9	6	4.62ml	

# Modelo Conceitual



# Modelo Lógico



# Modelo Físico

```
1  /*Eliminando estrutura do banco de dados*/
2  drop table if exists GRUPO_FAMILIAR cascade;
3  drop table if exists PESSOA cascade;
4  drop table if exists PESSOA_CONSOME cascade;
5  drop table if exists FORNECEDOR cascade;
6  drop table if exists FORNECEDOR_PRODUTO cascade;
7  drop table if exists PRODUTO cascade;
8  drop table if exists TIPO_PRODUTO cascade;
9
10
11 /*Criacao da estrutura das tabelas*/
12 create table GRUPO_FAMILIAR(
13     cod_grupo_familiar SERIAL primary key not null,
14     nome_grupo_familiar varchar(30) not null
15 );
16
17
18 create table PESSOA(
19     cod_pessoa SERIAL primary key not null,
20     fk_cod_grupo_familiar integer not null,
21     nome_pessoa varchar(50) not null,
22     cpf varchar(11) not null,
23     data_nascimento date not null
24 );
25
26 create table PESSOA_CONSOME(
27     cod_pessoa_consome SERIAL primary key not null,
28     fk_cod_produto integer not null,
29     fk_cod_pessoa integer not null,
30     quantidade_produto float not null
31 );
```

```
33 create table FORNECEDOR(
34     cod_fornecedor SERIAL primary key not null,
35     nome_fornecedor varchar(30) not null
36 );
37
38 create table FORNECEDOR_PRODUTO(
39     cod_fornecedor_produto SERIAL primary key not null,
40     fk_cod_fornecedor integer not null,
41     fk_cod_produto integer not null
42 );
43
44 create table PRODUTO(
45     cod_produto SERIAL primary key not null,
46     fk_cod_tipo_produto integer not null,
47     preco_produto float not null,
48     nome_produto varchar(40) not null
49 );
50
51 create table TIPO_PRODUTO(
52     cod_tipo_produto SERIAL primary key not null,
53     nome_tipo_produto varchar(30) not null
54 );
55
56 alter table PESSOA
57 add foreign key(fk_cod_grupo_familiar) references GRUPO_FAMILIAR(cod_grupo_familiar);
58
59 alter table PESSOA_CONSOME
60 add foreign key(fk_cod_produto) references PRODUTO(cod_produto),
61 add foreign key(fk_cod_pessoa) references PESSOA(cod_pessoa);
62
63 alter table FORNECEDOR_PRODUTO
64 add foreign key(fk_cod_fornecedor) references FORNECEDOR(cod_fornecedor),
65 add foreign key(fk_cod_produto) references PRODUTO(cod_produto);
66
67 alter table PRODUTO
68 add foreign key(fk_cod_tipo_produto) references TIPO_PRODUTO(cod_tipo_produto);
```



# Insert dos dados








```
70 /*Insercao dos dados*/
71 insert into grupo_familiar (nome_grupo_familiar)
72 values ('Minha Casa'),
73        ('Familia de pescadores - VV'),
74        ('Taberna dos viajantes - ANDO');
75
76
77 insert into pessoa(fk_cod_grupo_familiar, nome_pessoa, cpf, data_nascimento)
78 values (1, 'Kelvin L', '83087427853', '1998-08-15'),
79        (1, 'Valdeci G', '61576870880', '1977-09-03'),
80        (1, 'Clara L', '80663532787', '2007-12-20'),
81        (1, 'Kelen L', '6410440979', '1979-07-05'),
82        (2, 'Sebastian da Silva', '69505791607', '1986-12-02'),
83        (2, 'Eliot S', '14101087830', '1988-01-28'),
84        (2, 'Tião Moreira', '92251924876', '1983-05-03'),
85        (2, 'Haley Quinn', '2407861736', '1996-06-05'),
86        (2, 'Nair P', '27148449286', '1993-05-29'),
87        (2, 'Fabio Brazza', '10176554130', '1998-09-08'),
88        (3, 'Alexander Kernel', '43490663063', '1990-09-06'),
89        (3, 'Rita da Conquista', '73490663063', '1996-08-26'),
90        (3, 'Arthur Excalibur', '46733400006', '1998-07-30'),
91        (3, 'Morgan Freeman', '09248073000', '1961-05-08'),
92        (3, 'Gerald de Rivia', '49033310082', '1934-10-09'),
93        (3, 'Merlin de Avalon', '93397782007', '1960-05-17'),
94        (3, 'Gawyn Cristan', '11873481047', '1995-07-27'),
95        (3, 'Borz Graal', '16521580060', '1955-08-21'),
96        (3, 'Mordred Moreira', '63320779052', '1955-08-21'),
97        (3, 'Derfel da Silva', '88385078053', '1923-12-31');
98
99
100 insert into fornecedor(nome_fornecedor)
101 values ('MegaG'),
102        ('Ryu Alimentos'),
103        ('G2F'),
104        ('Distrimed'),
105        ('Famarcia do povo'),
106        ('Alimentos CIA'),
107        ('Limpeza do dia');
108
```

```
110 insert into tipo_produto(nome_tipo_produto)
111 values ('Alimenticio'),
112        ('Medicinal'),
113        ('Limpeza Geral'),
114        ('Limpeza Pessoal');
115
116
117 insert into produto(fk_cod_tipo_produto, nome_produto, preco_produto)
118 values (1, 'Arroz', 7.79),
119        (1, 'Feijão', 5.99),
120        (4, 'Pasta de dente', 3.79),
121        (2, 'Antibiotico', 17.03),
122        (1, 'Peixe', 37.90),
123
124
125 insert into fornecedor_produto(fk_cod_fornecedor, fk_cod_produto)
126 values (1,1),
127        (1,2),
128        (2,5),
129        (3,3),
130        (3,6),
131        (4,4),
132        (5,10),
133
134 insert into pessoa_consoma(fk_cod_pessoa, fk_cod_produto, quantidade_produto)
135 values (1,1,300.0),
136        (2,2,200.0),
137        (3,4,1),
138        (4,3,1),
139        (3,5,100),
140        (2,6,250),
141        (4,5,100),
142        (1,2,200),
143        (2,1,200),
144        (5,7,22),
145        (6,8,100),
```

# Consultas com Inner Join e Order By

```
SELECT grupo_familiar.nome_grupo_familiar, pessoa.nome_pessoa,
pessoa_consome.cod_pessoa_consome, produto.nome_produto,
tipo_produto.nome_tipo_produto, fornecedor_produto.cod_fornecedor_produto, fornecedor.nome_fornecedor
FROM pessoa
INNER JOIN grupo_familiar
ON(grupo_familiar.cod_grupo_familiar = pessoa.fk_cod_grupo_familiar)
INNER JOIN pessoa_consome
ON(pessoa_consome.fk_cod_pessoa = pessoa.cod_pessoa)
INNER JOIN produto
ON(produto.cod_produto = pessoa_consome.fk_cod_produto)
INNER JOIN tipo_produto
ON(tipo_produto.cod_tipo_produto = produto.fk_cod_tipo_produto)
INNER JOIN fornecedor_produto
ON(fornecedor_produto.fk_cod_produto = produto.cod_produto)
INNER JOIN fornecedor
ON(fornecedor.cod_fornecedor = fornecedor_produto.fk_cod_fornecedor)
ORDER BY grupo_familiar.nome_grupo_familiar, pessoa.nome_pessoa;
```

grupo\_familiar(+)  


 SELECT grupo\_familiar.nome\_grupo\_familiar, pessoa.r  Enter a SQL expression to filter results (use Ctrl+Space)     

Grid	ABC nome_grupo_familiar	ABC nome_pessoa	123 cod_pessoa_consome	ABC nome_produto	ABC nome_tipo_produto	123 cod_fornecedor_produto	ABC nome_fornecedor
10	Familia de pescadores - VV	Nair P	20	Arroz	Alimenticio	1	MegaG
11	Familia de pescadores - VV	Nair P	43	Filé de Frango	Alimenticio	24	Ryu Alimentos
12	Familia de pescadores - VV	Nair P	32	Carne de porco	Alimenticio	13	Ryu Alimentos
13	Familia de pescadores - VV	Nair P	14	Insulina	Medicinal	12	Distrimed
14	Familia de pescadores - VV	Sebastian da Silva	16	Arroz	Alimenticio	1	MegaG
15	Familia de pescadores - VV	Sebastian da Silva	10	Macarrão	Alimenticio	8	Alimentos CIA
16	Familia de pescadores - VV	Sebastian da Silva	39	Tomate Carmem	Alimenticio	20	MegaG
17	Familia de pescadores - VV	Tião Moreira	18	Feijão	Alimenticio	2	MegaG
18	Familia de pescadores - VV	Tião Moreira	12	Enxak	Medicinal	11	Alimentos CIA
19	Familia de pescadores - VV	Tião Moreira	41	Batata Monalise	Alimenticio	22	G2F
20	Minha Casa	Clara L	3	Antibiotico	Medicinal	6	Distrimed
21	Minha Casa	Clara L	57	Alcool em gel	Limpeza Pessoal	9	Limpeza do dia
22	Minha Casa	Clara L	37	Água sanitária	Limpeza Geral	19	G2F
23	Minha Casa	Clara L	5	Peixe	Alimenticio	3	Ryu Alimentos
24	Minha Casa	Kelen L	7	Peixe	Alimenticio	3	Ryu Alimentos
25	Minha Casa	Kelen L	38	Presunto Fatiado	Alimenticio	21	Ryu Alimentos
26	Minha Casa	Kelen L	4	Pasta de dente	Limpeza Pessoal	4	G2F
27	Minha Casa	Kelvin L	55	Maça	Alimenticio	7	Farmacia do povo



# Consultas com Inner Join e Order By

```
SELECT fornecedor.nome_fornecedor, produto.nome_produto as fornece
FROM produto
INNER JOIN fornecedor_produto
ON (produto.cod_produto = fornecedor_produto.fk_cod_produto)
INNER JOIN fornecedor
ON (fornecedor_produto.fk_cod_fornecedor = fornecedor.cod_fornecedor)
ORDER BY nome_fornecedor;
```


fornecedor(+)  Enter a SQL expression to filter results (use Ctrl+Sp

Grid

	ABC nome_fornecedor	ABC fornece
1	Alimentos CIA	Macarrão
2	Alimentos CIA	Enxak
3	Distrimed	Insulina
4	Distrimed	Antibiotico
5	Farmacia do povo	Maça
6	Farmacia do povo	Xarope
7	G2F	Shampoo
8	G2F	Pasta de dente
9	G2F	Água sanitária
10	G2F	Desinfetante
11	G2F	Ovos brancos
12	G2F	Batata Monalise
13	Limpeza do dia	Sacos de lixo
14	Limpeza do dia	Alcool em gel
15	MegaG	Arroz
16	MegaG	Feijão
17	MegaG	Farinha
18	MegaG	Leite condensado
19	MegaG	Tomate Carmem
20	MegaG	Dorflex
21	Ryu Alimentos	Presunto Fatiado
22	Ryu Alimentos	Carne de porco
23	Ryu Alimentos	Peixe
24	Ryu Alimentos	Filé de Frango
25	Ryu Alimentos	Sabão em pó
26	Ryu Alimentos	Vitamina C

Record

```
SELECT DISTINCT fornecedor.nome_fornecedor, tipo_produto.nome_tipo_produto as fornece_tipo
FROM produto
INNER JOIN fornecedor_produto
ON (produto.cod_produto = fornecedor_produto.fk_cod_produto)
INNER JOIN fornecedor
ON (fornecedor_produto.fk_cod_fornecedor = fornecedor.cod_fornecedor)
INNER JOIN tipo_produto
ON (tipo_produto.cod_tipo_produto = produto.fk_cod_tipo_produto)
ORDER BY nome_fornecedor;
```

fornecedor(+)  Enter a SQL expression to filter results (use Ctrl+Space)

Grid

	ABC nome_fornecedor	ABC fornece_tipo
1	Alimentos CIA	Alimenticio
2	Alimentos CIA	Medicinal
3	Distrimed	Medicinal
4	Farmacia do povo	Alimenticio
5	Farmacia do povo	Medicinal
6	G2F	Alimenticio
7	G2F	Limpeza Geral
8	G2F	Limpeza Pessoal
9	Limpeza do dia	Limpeza Geral
10	Limpeza do dia	Limpeza Pessoal
11	MegaG	Alimenticio
12	MegaG	Medicinal
13	Ryu Alimentos	Alimenticio
14	Ryu Alimentos	Limpeza Geral
15	Ryu Alimentos	Medicinal

# Consultas com Inner Join e Order By

```
SELECT DISTINCT pessoa.nome_pessoa, tipo_produto.nome_tipo_produto as consome_tipo
FROM produto
INNER JOIN pessoa_consome
ON(produto.cod_produto = pessoa_consome.fk_cod_produto)
INNER JOIN pessoa
ON(pessoa_consome.fk_cod_pessoa = pessoa.cod_pessoa)
INNER JOIN tipo_produto
ON(tipo_produto.cod_tipo_produto = produto.fk_cod_tipo_produto)
ORDER BY nome_pessoa desc;
```

pessoa(+)

SELECT DISTINCT pessoa.nome\_pessoa, tipo\_produto.

	ABC nome_pessoa	ABC consome_tipo
1	Valdeci G	Alimenticio
2	Valdeci G	Limpeza Geral
3	Valdeci G	Medicinal
4	Tião Moreira	Alimenticio
5	Tião Moreira	Medicinal
6	Sebastian da Silva	Alimenticio
7	Rita da Conquista	Alimenticio
8	Rita da Conquista	Limpeza Pessoal
9	Nair P	Alimenticio
10	Nair P	Medicinal
11	Morgan Freeman	Limpeza Pessoal
12	Mordred Moreira	Alimenticio
13	Merlin de Avalon	Alimenticio
14	Kelvin L	Alimenticio
15	Kelen L	Alimenticio
16	Kelen L	Limpeza Geral
17	Kelen L	Limpeza Pessoal
18	Haley Quinn	Alimenticio
19	Haley Quinn	Limpeza Geral
20	Gerald de Rivia	Alimenticio
21	Gerald de Rivia	Medicinal
22	Gawayn Cristan	Alimenticio
23	Gawayn Cristan	Limpeza Geral
24	Fabio Brazza	Alimenticio

```
SELECT DISTINCT produto.nome_produto as distribuir, grupo_familiar.nome_grupo_familiar
FROM produto
INNER JOIN pessoa_consome
ON(produto.cod_produto = pessoa_consome.fk_cod_produto)
INNER JOIN pessoa
ON(pessoa_consome.fk_cod_pessoa = pessoa.cod_pessoa)
INNER JOIN grupo_familiar
ON(grupo_familiar.cod_grupo_familiar = pessoa.fk_cod_grupo_familiar)
ORDER BY distribuir;
```

produto(+)

SELECT DISTINCT produto.nome\_produto as distribuir,

	ABC distribuir	ABC nome_grupo_familiar
1	Água sanitária	Minha Casa
2	Alcool em gel	Familia de pescadores - VV
3	Alcool em gel	Minha Casa
4	Antibiotico	Minha Casa
5	Antibiotico	Taberna dos viajantes - ANDO
6	Arroz	Familia de pescadores - VV
7	Arroz	Minha Casa
8	Arroz	Taberna dos viajantes - ANDO
9	Batata Monalise	Familia de pescadores - VV
10	Café	Familia de pescadores - VV
11	Carne de porco	Familia de pescadores - VV
12	Carne de porco	Taberna dos viajantes - ANDO
13	Cloro	Familia de pescadores - VV
14	Cloro	Minha Casa
15	Desinfetante	Familia de pescadores - VV
16	Desinfetante	Minha Casa
17	Desinfetante	Taberna dos viajantes - ANDO
18	Dorflex	Familia de pescadores - VV
19	Enxak	Familia de pescadores - VV
20	Enxak	Taberna dos viajantes - ANDO
21	Farinha	Familia de pescadores - VV
22	Farinha	Taberna dos viajantes - ANDO
23	Feijão	Familia de pescadores - VV
24	Feijão	Minha Casa

# Consulta com Group By e Funções de Agrupamento

SELECT pessoa.nome\_pessoa, sum(produto.preco\_produto) as soma\_valor\_produtos  
FROM pessoa  
INNER JOIN pessoa\_consome  
ON(pessoa\_consome.fk\_cod\_pessoa = pessoa.cod\_pessoa)  
INNER JOIN produto  
on(produto.cod\_produto = pessoa\_consome.fk\_cod\_produto)  
GROUP BY pessoa.nome\_pessoa  
ORDER BY soma\_valor\_produtos;

Grid

	nome_pessoa	soma_valor_produtos
1	Morgan Freeman	3.79
2	Borz Graal	4.39
3	Arthur Excalibur	8.49
4	Gawayn Cristan	8.58
5	Mordred Moreira	10.78
6	Haley Quinn	11.67
7	Derfel da Silva	15.38
8	Rita da Conquista	16.13
9	Sebastian da Silva	19.76
10	Tião Moreira	20.83
11	Fabio Brazza	21.66
12	Kelvin L	22.11
13	Alexander Kernel	26.28
14	Eliot S	30.97
15	Valdeci G	31.26
16	Merlin de Avalon	40.28
17	Nair P	44.67
18	Gerald de Rivia	54.93
19	Kelen L	61.28
20	Clara L	78.21

SELECT fornecedor.nome\_fornecedor,  
count(produto.fk\_cod\_tipo\_produto) as qtd\_produtos\_fornecidos  
FROM fornecedor  
INNER JOIN fornecedor\_produto  
ON(fornecedor.cod\_fornecedor = fornecedor\_produto.fk\_cod\_fornecedor)  
INNER JOIN produto  
ON(produto.cod\_produto = fornecedor\_produto.fk\_cod\_produto)  
GROUP BY fornecedor.nome\_fornecedor  
ORDER BY fornecedor.nome\_fornecedor;

Grid

	nome_fornecedor	qtd_produtos_fornecidos
1	Alimentos CIA	2
2	Distrimed	2
3	Farmacia do povo	2
4	G2F	6
5	Limpeza do dia	2
6	MegaG	6
7	Ryu Alimentos	6

# Consulta com Group By e Funções de Agrupamento

```
SELECT pessoa.nome_pessoa, count(*) as consome, tipo_produto.nome_tipo_produto
FROM pessoa
INNER JOIN pessoa_consome
ON(pessoa_consome.fk_cod_pessoa = pessoa.cod_pessoa)
INNER JOIN produto
on(produto.cod_produto = pessoa_consome.fk_cod_produto)
INNER JOIN tipo_produto
ON(produto.fk_cod_tipo_produto = tipo_produto.cod_tipo_produto)
GROUP BY pessoa.nome_pessoa, tipo_produto.cod_tipo_produto
ORDER BY nome_pessoa;
```

pessoa(+)

SELECT pessoa.nome\_pessoa, count(\*) as consome, tipo\_produto.nome\_tipo\_produto

	nome_pessoa	consome	nome_tipo_produto
1	Alexander Kernel	2	Alimenticio
2	Alexander Kernel	1	Medicinal
3	Arthur Excalibur	1	Alimenticio
4	Arthur Excalibur	1	Limpeza Geral
5	Borz Graal	1	Alimenticio
6	Clara L	1	Alimenticio
7	Clara L	1	Medicinal
8	Clara L	1	Limpeza Geral
9	Clara L	1	Limpeza Pessoal
10	Derfel da Silva	1	Alimenticio
11	Derfel da Silva	1	Medicinal
12	Eliot S	3	Alimenticio
13	Eliot S	1	Limpeza Geral
14	Fabio Brazza	1	Alimenticio
15	Fabio Brazza	1	Medicinal
16	Fabio Brazza	1	Limpeza Pessoal
17	Gawayn Cristan	1	Alimenticio
18	Gawayn Cristan	1	Limpeza Geral
19	Gerald de Rivia	1	Alimenticio
20	Gerald de Rivia	1	Medicinal
21	Haley Quinn	2	Alimenticio
22	Haley Quinn	1	Limpeza Geral
23	Kelen L	2	Alimenticio

# Consulta com Group By e Funções de Agrupamento

```
SELECT tipo_produto.nome_tipo_produto, count(*)
FROM tipo_produto
INNER JOIN produto
ON(produto.fk_cod_tipo_produto = tipo_produto.cod_tipo_produto)
GROUP BY(tipo_produto.cod_tipo_produto);
```

tipo\_produto

SELECT tipo\_produto.nome\_tipo\_produto, count(\*)

	ABC nome_tipo_produto	count
1	Limpeza Pessoal	3
2	Medicinal	6
3	Limpeza Geral	6
4	Alimenticio	15

```
SELECT produto.nome_produto, count(*) as qtd_pessoas_consumindo
FROM produto
INNER JOIN pessoa_consome
ON(pessoa_consome.fk_cod_produto = produto.cod_produto)
GROUP BY produto.nome_produto;
ORDER BY produto.nome_produto;
```

produto

SELECT produto.nome\_produto, count(\*) as qtd\_pess

	ABC nome_produto	qtd_pessoas_consumindo
1	Água sanitária	1
2	Alcool em gel	2
3	Antibiotico	2
4	Arroz	6
5	Batata Monalisa	1
6	Café	1
7	Carne de porco	2
8	Cloro	2
9	Desinfetante	3
10	Dorflex	1
11	Enxak	2
12	Farinha	3
13	Feijão	7
14	Filé de Frango	1
15	Insulina	2
16	Leite condensado	1
17	Maçã	3
18	Macarrão	3
19	Ovos brancos	1
20	Pasta de dente	3
21	Peixe	4
22	Presunto Fatiado	2
23	Sabão em pó	1
24	Sacos de lixo	1
25	Sopa enlatada	1
26	Tomate Carmem	1
27	Vitamina C	1



# Consultas com Left e Full Join

SELECT nome\_produto, nome\_fornecedor  
FROM produto  
LEFT JOIN fornecedor\_produto  
ON(fornecedor\_produto.fk\_cod\_produto = produto.cod\_produto)  
LEFT JOIN fornecedor  
ON(fornecedor.cod\_fornecedor = fornecedor\_produto.fk\_cod\_fornecedor);

produto(+)

SELECT nome\_produto, nome\_fornecedor FROM prod

	ABC nome_produto	ABC nome_fornecedor
6	Antibiotico	Distrimed
7	Maçã	Farmacia do povo
8	Macarrão	Alimentos CIA
9	Alcool em gel	Limpeza do dia
10	Farinha	MegaG
11	Enxak	Alimentos CIA
12	Insulina	Distrimed
13	Carne de porco	Ryu Alimentos
14	Sacos de lixo	Limpeza do dia
15	Shampoo	G2F
16	Xarope	Farmacia do povo
17	Leite condensado	MegaG
18	Sabão em pó	Ryu Alimentos
19	Água sanitária	G2F
20	Tomate Carmem	MegaG
21	Presunto Fatiado	Ryu Alimentos
22	Batata Monalise	G2F
23	Ovos brancos	G2F
24	Filé de Frango	Ryu Alimentos
25	Dorflex	MegaG
26	Vitamina C	Ryu Alimentos
27	Sopa enlatada	[NULL]
28	Sabonete	[NULL]
29	Cloro	[NULL]
30	Café	[NULL]

SELECT produto.nome\_produto, pessoa\_consome.fk\_cod\_pessoa, fornecedor\_produto.fk\_cod\_fornecedor  
FROM produto  
FULL JOIN pessoa\_consome  
ON(pessoa\_consome.fk\_cod\_produto = produto.cod\_produto)  
FULL JOIN fornecedor\_produto  
ON(fornecedor\_produto.fk\_cod\_produto = produto.cod\_produto);

produto(+)

fornecedor cod\_fornecedor = 4

	ABC nome_produto	123 fk_cod_pessoa	123 fk_cod_fornecedor
37	Enxak	20	6
38	Enxak	7	6
39	Insulina	2	4
40	Insulina	9	4
41	Carne de porco	9	2
42	Carne de porco	11	2
43	Sacos de lixo	13	7
44	Shampoo	[NULL]	3
45	Xarope	[NULL]	5
46	Leite condensado	1	1
47	Sabão em pó	2	2
48	Água sanitária	3	3
49	Tomate Carmem	5	1
50	Presunto Fatiado	6	2
51	Presunto Fatiado	4	2
52	Batata Monalise	7	3
53	Ovos brancos	8	3
54	Filé de Frango	9	2
55	Dorflex	10	1
56	Vitamina C	11	2
57	Sopa enlatada	12	[NULL]
58	Sabonete	[NULL]	[NULL]
59	Cloro	4	[NULL]
60	Cloro	6	[NULL]
61	Café	5	[NULL]

# Consultas que utilize View com Subconsulta

```
create or replace view pessoa_demandam_medicinais AS
SELECT pessoa.nome_pessoa
FROM pessoa
WHERE pessoa.cod_pessoa IN
( SELECT pessoa_consomme.fk_cod_pessoa
  FROM pessoa_consomme
  INNER JOIN produto
  ON(produto.cod_produto = pessoa_consomme.fk_cod_produto)
  WHERE produto.fk_cod_tipo_produto = 2);

select * from pessoa_demandam_medicinais;
```

Grid

	nome_pessoa
1	Clara L
2	Derfel da Silva
3	Alexander Kernel
4	Fabio Brazza
5	Nair P
6	Tião Moreira
7	Valdeci G
8	Gerald de Rivia

```
create or replace view PRODUTOS_NAO_DEMANDADOS AS
SELECT produto.nome_produto
FROM produto
WHERE produto.cod_produto NOT IN
( SELECT pessoa_consomme.fk_cod_produto
  FROM pessoa_consomme );

select * from PRODUTOS_NAO_DEMANDADOS;
```

Grid

	nome_produto
1	Sabonete
2	Xarope
3	Shampoo

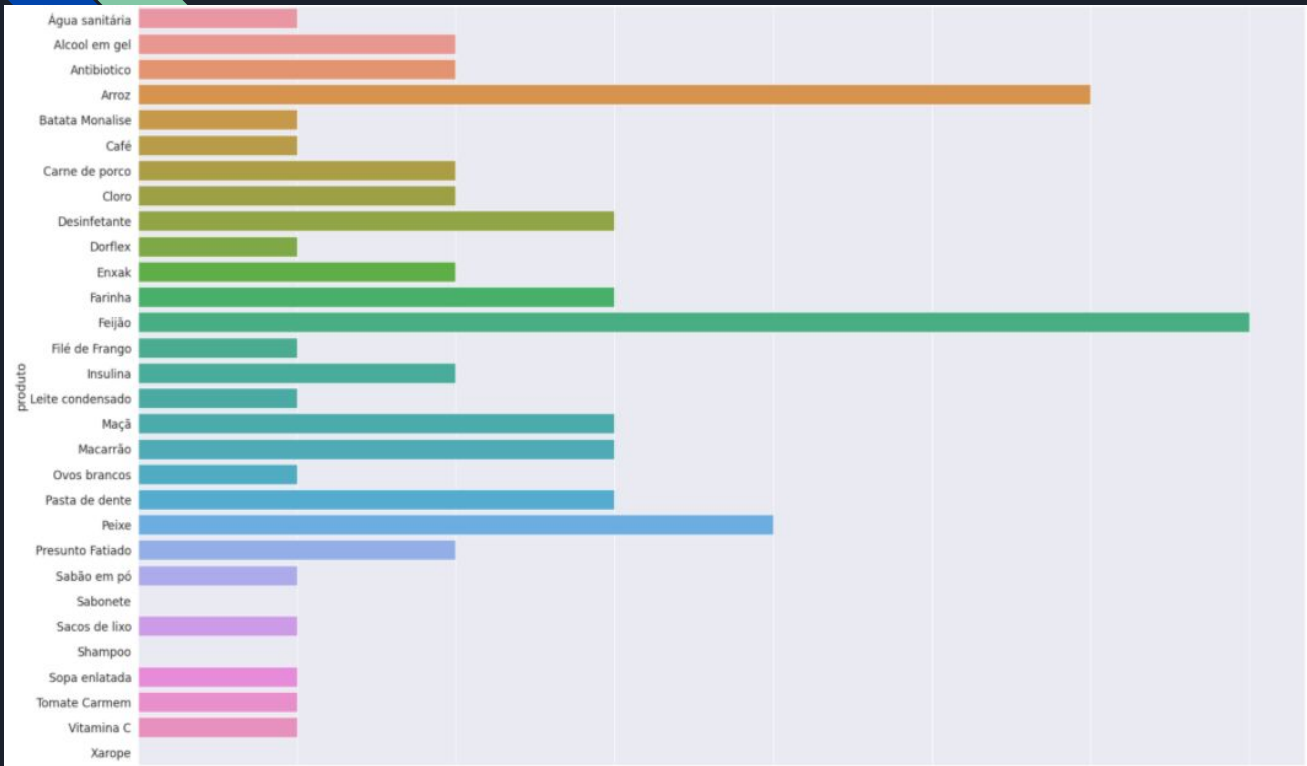


# Perguntas a serem respondidas

- Quais os produtos mais requisitados?
- Quais fornecedores tem mais variedade de produtos ofertados?
- Qual o gasto de cada família?
- Quais famílias tendem a comprar produtos de determinados preços?
- Qual a tendência de idade dos consumidores de cada produto?



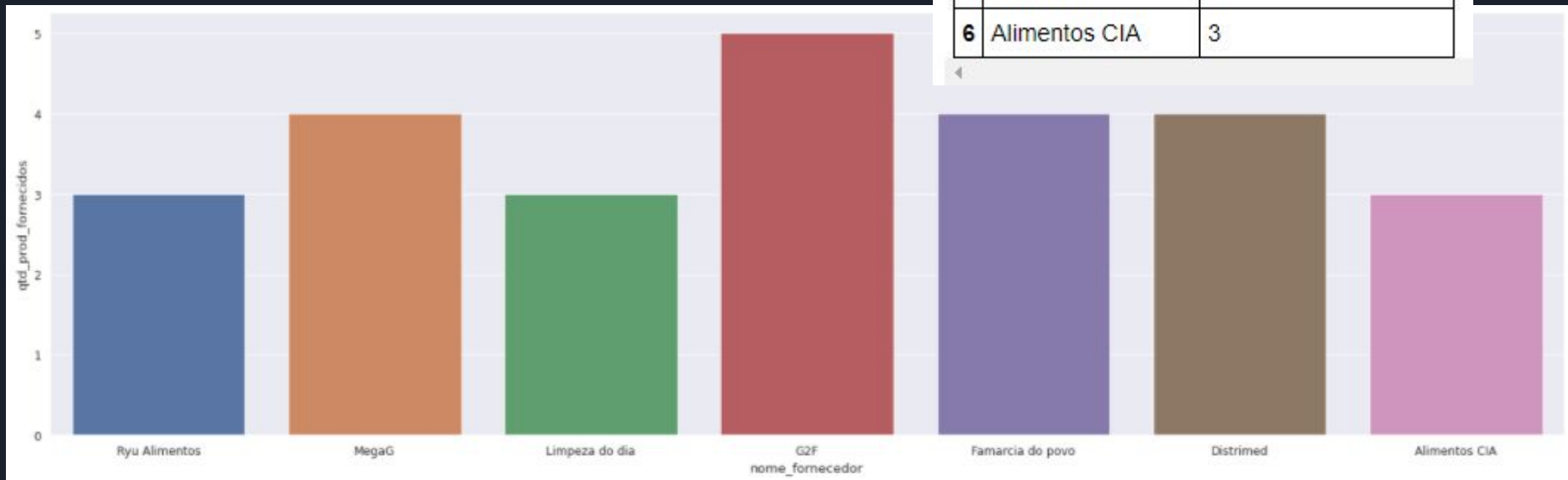
# Relatórios e Gráficos



	qtd_consumidores	produto
0	1	Água sanitária
1	2	Alcool em gel
2	2	Antibiotico
3	6	Arroz
4	1	Batata Monalise
5	1	Café
6	2	Carne de porco
7	2	Cloro
8	3	Desinfetante
9	1	Dorflex
10	2	Enxak
11	3	Farinha
12	7	Feijão
13	1	Filé de Frango
14	2	Insulina
15	1	Leite condensado
16	3	Maçã
17	3	Macarrão
18	1	Ovos brancos
19	3	Pasta de dente
20	4	Peixe
21	2	Presunto Fatiado
22	1	Sabão em pó
23	0	Sabonete
24	1	Sacos de lixo
25	0	Shampoo

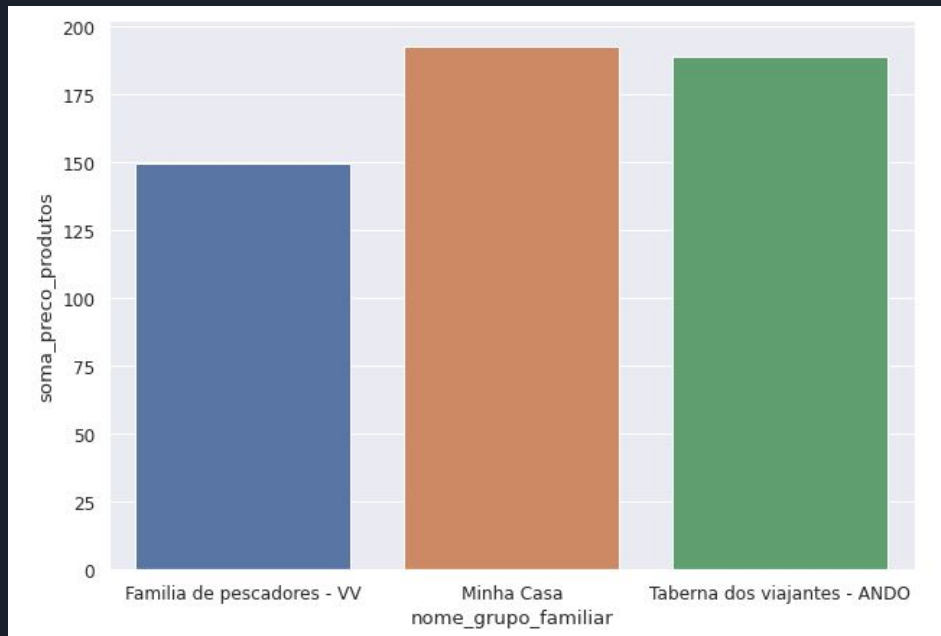
# Relatórios e Gráficos

	nome_fornecedor	qtd_prod_fornecidos
0	Ryu Alimentos	3
1	MegaG	4
2	Limpeza do dia	3
3	G2F	5
4	Famarcia do povo	4
5	Distrimed	4
6	Alimentos CIA	3



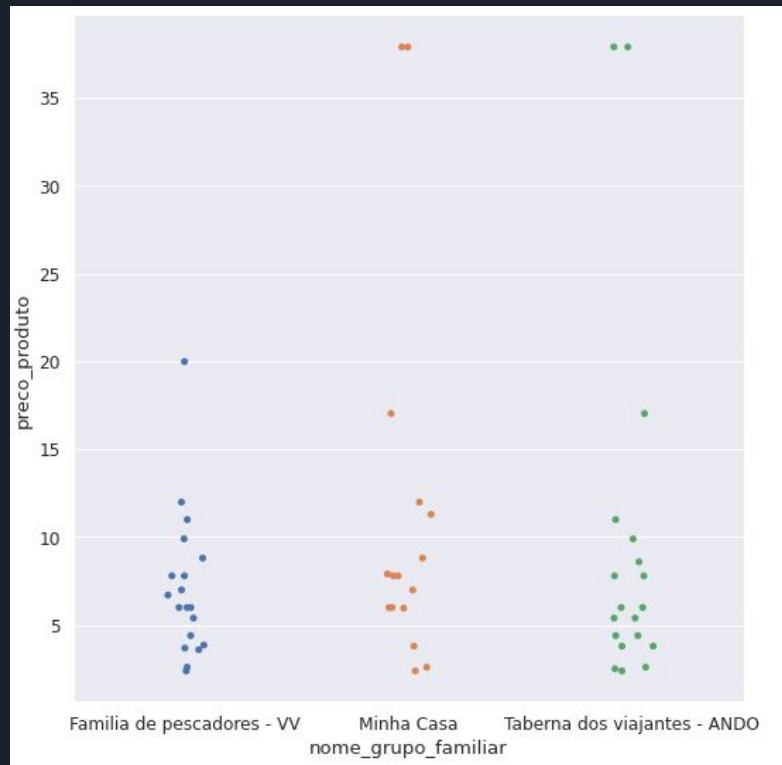
# Relatórios e Gráficos

	nome_grupo_familiar	soma_preco_produtos
0	Familia de pescadores - VV	149.56
1	Minha Casa	192.86
2	Taberna dos viajantes - ANDO	189.03



# Relatórios e Gráficos

	nome_grupo_familiar	nome_pessoa	nome_produto	preco_produto	nome_tipo_produto	nome_fornecedor
0	Familia de pescadores - VV	Eliot S	Farinha	5.39	Alimenticio	MegaG
1	Familia de pescadores - VV	Eliot S	Presunto Fatiado	8.80	Alimenticio	Famarcia do povo
2	Familia de pescadores - VV	Eliot S	Feijão	5.99	Alimenticio	MegaG
3	Familia de pescadores - VV	Fabio Brazza	Alcool em gel	11.99	Limpeza Pessoal	Limpeza do dia
4	Familia de pescadores - VV	Fabio Brazza	Feijão	5.99	Alimenticio	MegaG
5	Familia de pescadores - VV	Fabio Brazza	Dorflex	3.68	Medicinal	Distrimed
6	Familia de pescadores - VV	Haley Quinn	Maçã	2.38	Alimenticio	Famarcia do povo
7	Familia de pescadores - VV	Haley Quinn	Ovos brancos	6.70	Alimenticio	Limpeza do dia
8	Familia de pescadores - VV	Haley Quinn	Desinfetante	2.59	Limpeza Geral	G2F
9	Familia de pescadores - VV	Nair P	Arroz	7.79	Alimenticio	MegaG
10	Familia de pescadores - VV	Nair P	Filé de Frango	19.99	Alimenticio	Famarcia do povo
11	Familia de pescadores - VV	Nair P	Carne de porco	9.90	Alimenticio	Ryu Alimentos
12	Familia de pescadores - VV	Nair P	Insulina	6.99	Medicinal	Distrimed
13	Familia de pescadores - VV	Sebastian da Silva	Arroz	7.79	Alimenticio	MegaG
14	Familia de pescadores - VV	Sebastian da Silva	Macarrão	4.39	Alimenticio	Alimentos CIA
15	Familia de pescadores - VV	Sebastian da Silva	Tomate Carmem	3.59	Alimenticio	Distrimed
16	Familia de pescadores - VV	Tião Moreira	Feijão	5.99	Alimenticio	MegaG
17	Familia de pescadores - VV	Tião Moreira	Enxak	10.99	Medicinal	Alimentos CIA
18	Familia de pescadores - VV	Tião Moreira	Batata Monalise	3.85	Alimenticio	Alimentos CIA
19	Minha Casa	Clara L	Antibiotico	17.03	Medicinal	Distrimed
20	Minha Casa	Clara L	Alcool em gel	11.99	Limpeza Pessoal	Limpeza do dia
21	Minha Casa	Clara L	Água sanitária	11.29	Limpeza Geral	G2F
22	Minha Casa	Clara L	Peixe	37.90	Alimenticio	Ryu Alimentos
23	Minha Casa	Kelen L	Peixe	37.90	Alimenticio	Ryu Alimentos



# Relatórios e Gráficos

	nome_pessoa	data_nascimento	idade	nome_produto	nome_tipo_produto
0	Alexander Kernel	1990-09-06	30.0	Vitamina C	Medicinal
1	Alexander Kernel	1990-09-06	30.0	Carne de porco	Alimenticio
2	Alexander Kernel	1990-09-06	30.0	Arroz	Alimenticio
3	Arthur Excalibur	1998-07-30	22.0	Sacos de lixo	Limpeza Geral
4	Arthur Excalibur	1998-07-30	22.0	Feijão	Alimenticio
5	Borz Graal	1955-08-21	65.0	Macarrão	Alimenticio
6	Clara L	2007-12-20	12.0	Água sanitária	Limpeza Geral
7	Clara L	2007-12-20	12.0	Antibiótico	Medicinal
8	Clara L	2007-12-20	12.0	Alcool em gel	Limpeza Pessoal
9	Clara L	2007-12-20	12.0	Peixe	Alimenticio
10	Derfel da Silva	1923-12-31	96.0	Macarrão	Alimenticio
11	Derfel da Silva	1923-12-31	96.0	Enxak	Medicinal
12	Eliot S	1988-01-28	32.0	Feijão	Alimenticio
13	Eliot S	1988-01-28	32.0	Farinha	Alimenticio
14	Eliot S	1988-01-28	32.0	Presunto Fatiado	Alimenticio
15	Fabio Brazza	1998-09-08	22.0	Feijão	Alimenticio
16	Fabio Brazza	1998-09-08	22.0	Alcool em gel	Limpeza Pessoal
17	Fabio Brazza	1998-09-08	22.0	Dorflex	Medicinal
18	Gawayn Cristan	1995-07-27	25.0	Desinfetante	Limpeza Geral
19	Gawayn Cristan	1995-07-27	25.0	Feijão	Alimenticio
20	Gerald de Rivia	1934-10-09	85.0	Peixe	Alimenticio
21	Gerald de Rivia	1934-10-09	85.0	Antibiótico	Medicinal
22	Haley Quinn	1996-06-05	24.0	Desinfetante	Limpeza Geral
23	Haley Quinn	1996-06-05	24.0	Ovos brancos	Alimenticio
24	Haley Quinn	1996-06-05	24.0	Maçã	Alimenticio

