

[illegible]

Cliente_Veiculo

SQL File 5* x

Limit to 1000 rows

1 • select * from cliente_veiculo

Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:

	CODIGO	CPF/CNPJ	PLACA
1	60197669000	KLE2124	
2	65191419090	KLC5438	
3	66863480077	KIR8980	
4	54042619606	KLH6098	
5	14261490340	KLY7880	
6	46736818001	KKK5233	
7	61868631826	KGE9587	
8	17144517554	KJQ0759	
9	83824694425	KLY4260	
10	80518223310	KKN9114	
11	93456228023	KLI7347	
12	14870567539	KGF6677	
13	23326687052	KIK8444	
*	NULL	NULL	NULL

Empresa

SQL File 5* x

Limit to 1000 rows

1 • select * from empresa

Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:

	CODEMPRESA	CNPJ	RAZAO SOCIAL	CODENDEREÇO	LOGOEMPRESA
1	42132091000169	AUTO MECÂNICA LTDA	18	https://www.sennaautomecanica.com.br/img/logo.png	
*	NULL	NULL	NULL	NULL	NULL

SQL File 5* x

Limit to 1000 rows

1 • `select * from endereco`

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: W

	CODENDEREÇO	RUA	NUMERO	BAIRRO	CIDADE	ESTADO	CEP	LATITUDE	LONGITUDE
▶ 1		João Batista do Nascimento	17	Bom Jesus	Serra Talhada	PE	56906330	-7.98922	-38.3093
2		Antônio Nunes de Souza	60	Cagep	Serra Talhada	PE	56909640	-7.97755	-38.2931
3		Maria Luiza Kehrlé Murato	197	AABB	Serra Talhada	PE	56912194	-7.9957	-38.2854
4		Maria da Glória Nogueira	177	Várzea	Serra Talhada	PE	56906530	-7.99877	-38.2958
5		Deputado Afrânio Ribeiro de Godoy	1321	Nossa Senhora da Penha	Serra Talhada	PE	56903390	-7.99114	-38.2945
6		Irnério Inácio	28	Nossa Senhora da Penha	Serra Talhada	PE	56903902	-7.99273	-38.2995
7		Antônio Terto	240	Nossa Senhora da Penha	Serra Talhada	PE	56912390	-7.99011	-38.2923
8		Travessa João de Souza Santos	111	Tancredo Neves	Serra Talhada	PE	56909225	-7.98308	-38.2778
9		Avenida Antônio Romão de Farias	355	Bom Jesus	Serra Talhada	PE	56906170	-7.98451	-38.3069
10		Doutor Sérgio Magalhães	718	Centro	Serra Talhada	PE	56903910	-7.99145	-38.3009
11		Praça Manoel Pereira Lins	887	São Cristovão	Serra Talhada	PE	56903590	-7.98889	-38.2921
12		Francisco Porfiro	1375	Caxixola	Serra Talhada	PE	56915050	-8.00029	-38.3053
13		Inocêncio Gomes de Andrade	839	Nossa Senhora da Penha	Serra Talhada	PE	56903580	-7.98896	-38.2945
14		Manoel Rodrigues de Carvalho	341	Várzea	Serra Talhada	PE	56912300	-7.9979	-38.2863
15		Joaquim Alves de Magalhães	1530	AABB	Serra Talhada	PE	56912150	-7.9879	-38.2876
16		Coronel Cornélio Soares	474	Nossa Senhora da Penha	Serra Talhada	PE	56903970	-7.99216	-38.301
17		Cap. Antônio Inácio de Medeiros	1098	Bom Jesus	Serra Talhada	PE	56906340	-7.9886	-38.3102
18		Raimundo Ailton Rodrigues Santos	24	COHAB	Serra Talhada	PE	56909510	-7.9747	-38.2844
19		Professora Maria Luiza Duarte de Sá	6	Vila Bela	Serra Talhada	PE	56907050	-7.99695	-38.3136

[illegible]

Item

SQL File 5*

Limit to 1000 rows

1 • `select * from item`

Result Grid

	CODITEM	INSTRUCAO
▶	1	Verificar Óleo
	2	Levantar carro e escorrer o óleo
	3	Adicionar o novo óleo e descer o carro
	4	Inciar checklist
	5	Lubrificar todos parafusos
	6	Finalizar checklist
	7	Colocar macaco hidráulico e remover parafusos
	8	Remover pneu e colocar novo
	9	Colocar, apertar todos os parafusos e abaixar carro
	10	Iniciar Alinhamento e Balanceamento
	11	Verificar se tudo está de acordo com o checklist
	12	Finalizar Alinhamento e Balanceamento
	13	Iniciar revisão de acordo com o checklist
	14	Realizar limpeza do motor
	15	Finalizar revisão e validar checklist
*	NULL	NULL

Ordem_Servico

SQL File 5*

Limit to 1000 rows

1 • `select * from ordem_servico`

Result Grid

	CODIGO	CODORDEMSERVICO	CODSERVICO	CODFUNCIONARIO
▶	1	1	2	9
	2	2	4	10
	3	2	5	10
	4	3	4	1
	5	3	5	1
	6	4	2	1
	7	5	2	2
	8	6	4	2
	9	6	5	2
	10	7	4	10
	11	7	5	10
	12	8	4	10
	13	8	5	10
	14	9	4	1
	15	9	5	1
	16	10	2	9
	17	11	1	1
	18	12	1	1
	19	13	1	1
	20	14	2	10

ordem_servico 8

Ordemservice

1 • `select * from ordemservice`

	CODORDEMSERVICO	ENTRADA	SAIDA	TOTAL	CODCLIENTE	PLACA	CODFUNCIONARIO	PENDENTE	TOTALSERVICO	VALORDESCONTO
1		2021-11-08	2021-11-11	350	1	KLE2124	8	N	350	0
2		2021-11-01	2021-11-08	570	2	KLC5438	8	N	570	0
3		2021-10-25	2021-11-01	570	3	KIR8980	3	N	550	20
4		2021-10-25	2021-10-28	350	4	KLH6098	3	N	350	0
5		2021-10-25	2021-10-28	350	5	KLY7880	8	N	350	0
6		2021-10-18	2021-10-25	570	6	KKK5233	8	N	570	0
7		2021-10-18	2021-10-25	570	7	KGE9587	3	N	570	0
8		2021-10-04	2021-10-11	570	8	KJQ0759	8	N	570	0
9		2021-10-04	2021-10-11	570	9	KLY4260	3	N	570	0
10		2021-10-04	2021-10-07	350	10	KKN9114	3	N	330	20
11		2021-09-27	2021-09-27	65	11	KLI7347	8	N	65	0
12		2021-09-30	2021-09-30	65	12	KGF6677	8	N	65	0
13		2021-10-18	2021-10-18	65	13	KIK8444	8	N	65	0
14		2021-11-11	0000-00-00	350	11	KLI7347	3	S	350	0
15		2021-11-11	0000-00-00	400	12	KGF6677	8	S	400	0
16		2021-11-11	0000-00-00	170	13	KIK8444	3	S	170	0
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Servico

1 • `select * from servico`

	CODSERVICO	DESCRICAO	PRECO
1		Troca de Óleo	65
2		Revisão Geral	350
3		Troca de Pneu	20
4		Alinhamento e Balanceamento	400
5		Revisão do Motor	170
*	NULL	NULL	NULL

Servico_Item

SQL File 5* x

Limit to 1000 rows

```
1 • select * from servico_item
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

	CODIGO	CODSERVICO	CODITEM
▶	1	1	1
	2	1	2
	3	1	3
	4	2	4
	5	2	5
	6	2	6
	7	3	7
	8	3	8
	9	3	9
	10	4	10
	11	4	11
	12	4	12
	13	5	13
	14	5	14
	15	5	15
*	NULL	NULL	NULL

Veiculo

SQL File 5* x

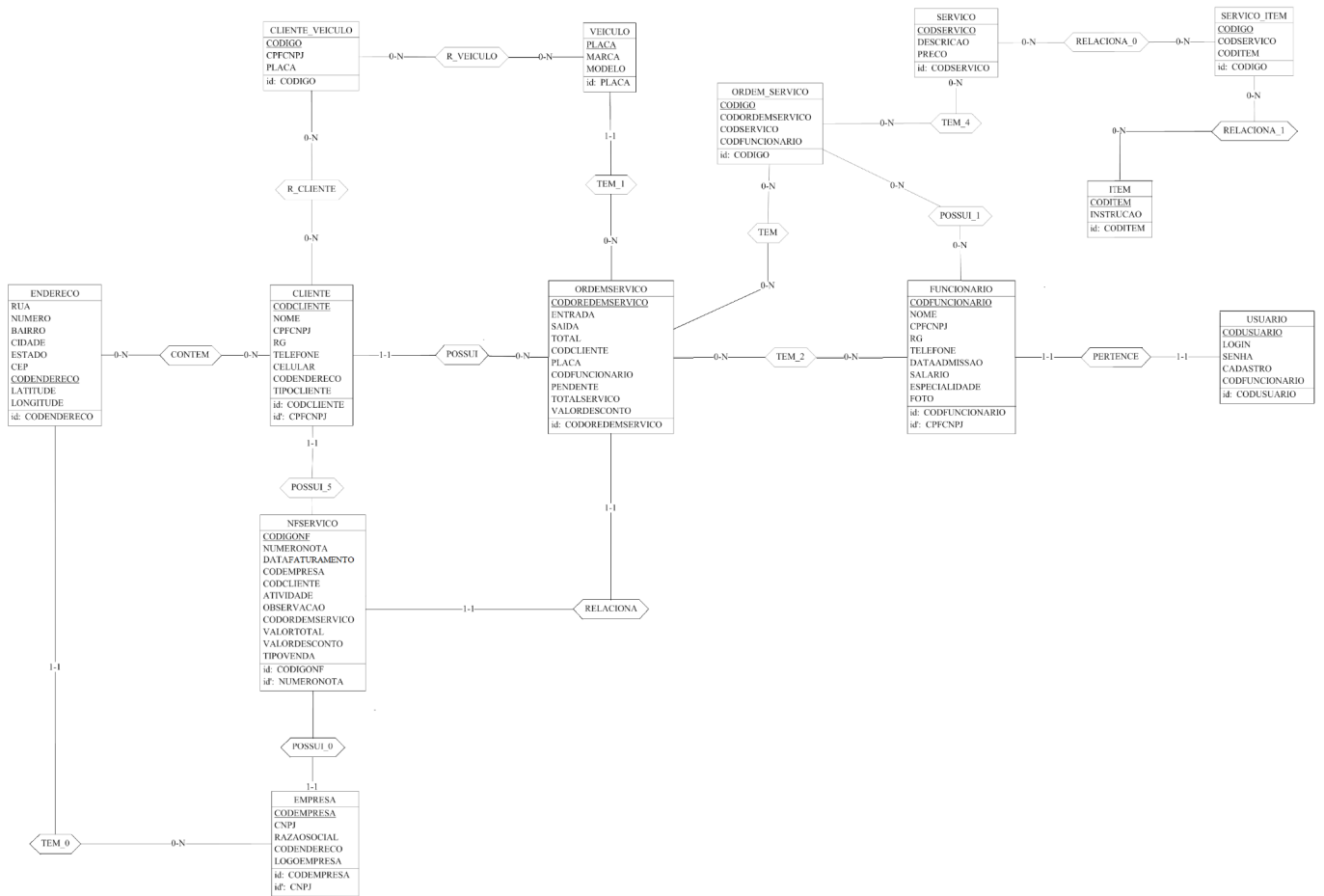
Limit to 1000 rows

```
1 • select * from veiculo
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

	PLACA	MARCA	MODELO
▶	KGE9587	Plymouth	Gran Voyager LE 2.5
	KGF6677	Cadillac	Deville/Eldorado 4.9
	KIK8444	MINI	COOPER 1.6 Mec.
	KIR8980	Buggy	Buggy 1.6 2-Lug.
	KJQ0759	Buggy	Buggy IV e V
	KKK5233	Ford	Corcel II L
	KKN9114	Rover	Mini Cooper 1.3
	KLC5438	Engesa	Engesa 4x4 4.0 Diesel
	KLE2124	Chrysler	TOWN & COUNTRY Touring 3.6 V6 Aut.
	KLH6098	Mahindra	SUV 2.2 4x4 (diesel) Mec.
	KLI7347	Porsche	911 Carrera Cabriolet 3.4/ 3.6 Mec
	KLY4260	Renault	19 RT 1.8/ 1.8i
	KLY7880	Mercury	Mystique GS 2.5 V6 Mec.
*	NULL	NULL	NULL

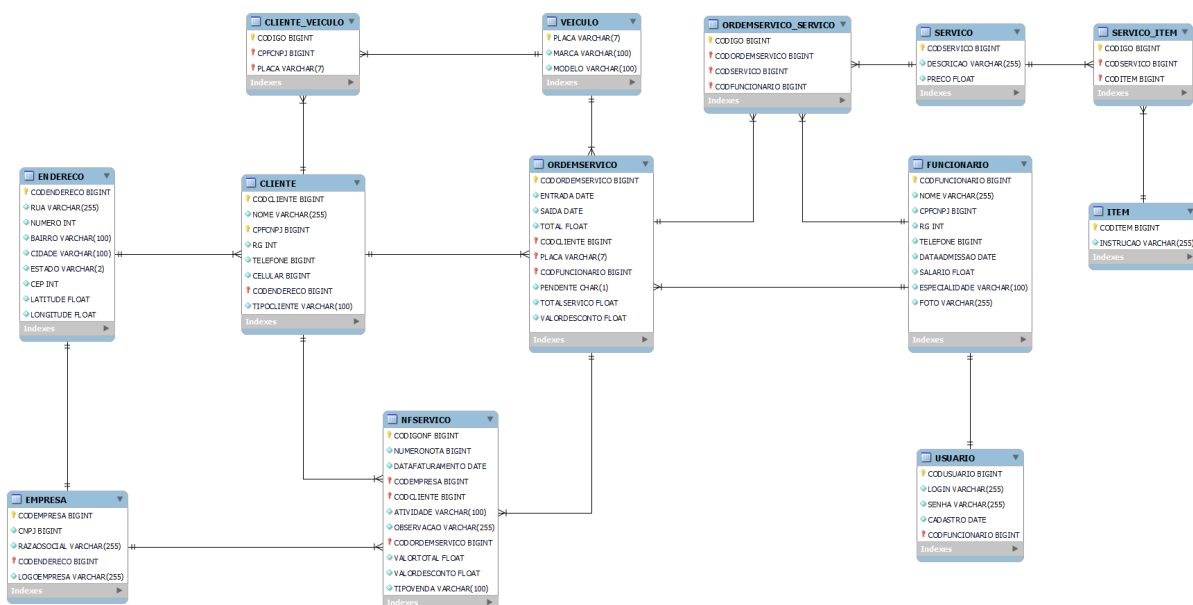
a) Diagrama de Entidades e Relacionamentos Conceitual



b) Modelo Lógico Relacional

1. CLIENTE (Codcliente, Nome, Cpfcpnpj, Rg, Telefone, Celular, Codendereco, Tipocliente);
2. CLIENTE_VEICULO (Codigo, Cpfcpnpj, Placa);
3. ENDERECO (Codendereco, Rua, Numero, Bairro, Cidade, Estado, Cep, Latitude, Longitude);
4. FUNCIONARIO (Codfuncionario, Nome, Cpfcpnpj, Rg, Telefone, Dataadmissao, Salario, Especialidade, Foto);
5. EMPRESA (Codempresa, Cnpj, Razaosocial, Codendereco, Logoempresa);
6. ITEM (Coditem, Instrucao);
7. SERVICO_ITEM (Codigo, Codservico, Coditem);
8. ORDEM_SERVICO (Codigo, Codordemservico, Codservico, Codfuncionario);
9. SERVICO (Codservico, Descricao, Preço);
10. NFSERVICO (Codigonf, Numeronota, Datafaturamento, Codempresa, Codcliente, Atividade, Observacao, Codordemservico, Valortotal, Valordesconto, Tipovenda);
11. ORDEMSERVICO (Codoredemservico, Entrada, Total, Codcliente, Placa, Codfuncionario, Pendente, Totalservico, Valordesconto);
12. USUARIO (Codusuario, Login, Senha, Cadastro, Codfuncionario);
13. VEICULO (PLACA, MARCA, MODELO);

c) Derivar o Modelo Lógico Relacional do BD



d) SQL do Modelo Relacional

```
-- Banco de dados

create database PBD_SACO;
use PBD_SACO;

-- Tabelas

create table CLIENTE (
    CODCLIENTE bigint not null auto_increment,
    NOME varchar(255) not null,
    CPFCNPJ bigint not null,
    RG int not null,
    TELEFONE bigint not null,
    CELULAR bigint not null,
    CODENDereco bigint not null,
    TIPOCLIENTE varchar(100) not null,
    constraint IDCLIENTE_1 unique (CPFCNPJ),
    constraint IDCLIENTE primary key (CODCLIENTE))
```

```
create table CLIENTE_VEICULO (
    CODIGO bigint not null auto_increment,
    CPFCNPJ bigint not null,
    PLACA varchar(7) not null,
    constraint IDCLIENTE_VEICULO primary key (CODIGO));

create table ENDERECO (
    CODENDERECO bigint not null auto_increment,
    RUA varchar(255) not null,
    NUMERO int not null,
    BAIRRO varchar(100) not null,
    CIDADE varchar(100) not null,
    ESTADO varchar(2) not null,
    CEP int not null,
    LATITUDE float not null,
    LONGITUDE float not null,
    constraint IDENDERECO primary key (CODENDERECO));

create table FUNCIONARIO (
    CODFUNCIONARIO bigint not null auto_increment,
    NOME varchar(255) not null,
    CPFCNPJ bigint not null,
    RG int not null,
    TELEFONE bigint not null,
    DATAADMISSAO date not null,
    SALARIO float not null,
    ESPECIALIDADE varchar(100) not null,
    FOTO varchar(255),
    constraint IDFUNCIONARIO_1 unique (CPFCNPJ),
    constraint IDFUNCIONARIO primary key (CODFUNCIONARIO));

create table EMPRESA (
    CODEMPRESA bigint not null auto_increment,
    CNPJ bigint not null,
    RAZAOSOCIAL varchar(255) not null,
    CODENDERECO bigint not null,
    LOGOEMPRESA varchar(255) not null,
    constraint IDEMPRESA_1 unique (CNPJ),
    constraint IDEMPRESA primary key (CODEMPRESA));
```

```
create table ITEM (  
    CODITEM bigint not null auto_increment,  
    INSTRUCAO varchar(255) not null,  
    constraint IDITEM primary key (CODITEM));  
  
create table SERVICO_ITEM (  
    CODIGO bigint not null auto_increment,  
    CODSERVICO bigint not null,  
    CODITEM bigint not null,  
    constraint IDSERVICO_ITEM primary key (CODIGO));  
  
create table ORDEM_SERVICO (  
    CODIGO bigint not null auto_increment,  
    CODORDEMSERVICO bigint not null,  
    CODSERVICO bigint not null,  
    CODFUNCIONARIO bigint not null,  
    constraint IDORDEM_SERVICO primary key (CODIGO));  
  
create table SERVICO (  
    CODSERVICO bigint not null auto_increment,  
    DESCRICAO varchar(255) not null,  
    PRECO float not null,  
    constraint IDSERVICO primary key (CODSERVICO));  
  
create table NFSERVICO (  
    CODIGONF bigint not null auto_increment,  
    NUMERONOTA bigint not null,  
    DATAFATURAMENTO date not null,  
    CODEMPRESA bigint not null,  
    CODCLIENTE bigint not null,  
    ATIVIDADE varchar(100) not null,  
    OBSERVACAO varchar(255) not null,  
    CODORDEMSERVICO bigint not null,  
    VALORTOTAL float not null,  
    VALORDESCONTO float not null,  
    TIPOVENDA varchar(100) not null,  
    constraint IDNFSERVICO primary key (CODIGONF),  
    constraint IDNFSERVICO_1 unique (NUMERONOTA));
```

```

create table ORDEMSERVICO (
    CODOREDEMSERVICO bigint not null auto_increment,
    ENTRADA date not null,
    SAIDA date not null,
    TOTAL float not null,
    CODCLIENTE bigint not null,
    PLACA varchar(7) not null,
    CODFUNCIONARIO bigint not null,
    PENDENTE char(1) not null,
    TOTALSERVICO float not null,
    VALORDESCONTO float not null,
    constraint IDORDEMSERVICO primary key (CODOREDEMSERVICO));

create table USUARIO (
    CODUSUARIO bigint not null auto_increment,
    LOGIN varchar(255) not null,
    SENHA varchar(255) not null,
    CADASTRO date not null,
    CODFUNCIONARIO bigint not null,
    constraint IDUSUARIO primary key (CODUSUARIO));

create table VEICULO (
    PLACA varchar(7) not null,
    MARCA varchar(100) not null,
    MODELO varchar(100) not null,
    constraint IDVEICULO primary key (PLACA));

```

e) Implementação do banco de dados no SGBD MySql.

Conforme solicitado foi criado o Database no MySql, via MySql Workbench, com comandos SQL e suas respectivas tabelas de acordo com a aplicação solicitada para o SACO.

f) Popular o banco de dados usando SQL.

Foram criados Scripts de Insert para todas as tabelas do banco de dados, esse processo trata-se de uma importação dos dados do cliente, geralmente quando o mesmo já possui outro sistema tecnológico esse processo torna-se bem mais prático. Já no caso da aplicação do SACO, é uma aplicação exclusiva, sendo assim, essa importação é mais trabalhosa, pois precisa ser transcrevida do papel para o banco de dados.

Exemplo de Script SQL de Insert para tabela de Cliente:

```
INSERT INTO CLIENTE
(`NOME`,
`CPF`CNPJ`,
`RG`,
`TELEFONE`,
`CELULAR`,
`CODENDEREÇO`,
`TIPOCLIENTE`)
VALUES
("Maria",
33333333333333,
33333333,
8199544687,
81999544687,
3,
"FREGUES");
```

Todos os Inserts realizados estão sendo enviados em um arquivo SQL junto ao projeto.

g) Consultar o banco de dados com as seguintes consultas típicas:

g-1) Quais são nossos Funcionários? Mostrar os dados do funcionário.

The screenshot shows the MySQL Workbench interface. The 'SQL File 2' window contains the query: `SELECT * FROM FUNCIONARIO`. The 'Result Grid' displays the following data:

CODFUNCIONARIO	NOME	CPF/CNPJ	RG	TELEFONE	DATAADMISSÃO	SALARIO	ESPECIALIDADE
1	Luís Erick Monteiro	14261490340	445235299	28266993	2014-05-31	1100	MECANICO
2	Marcelo Enrico Assis	17144517554	390515255	26325320	2020-03-03	1100	MECANICO
3	Lorenzo Thiago Nelson Santos	80518223310	390515255	39787452	2013-01-10	1700	SUPERVISOR DE MECA
4	Marcelo da Fonseca Souza	89100047031	162565422	8739964793	2013-01-10	3000	CHEFE DE OFICINA
5	Maria Luciana Pereira	93225451436	173176884	8739940683	2020-05-13	1300	ADMINISTRAÇÃO
6	José Claudio da Silva	66901237405	485471243	8725517942	2020-05-14	1300	FATURAMENTO
7	André Firmino Conceição	35524474405	151960574	8739061987	2020-05-17	1100	RECEPCIONISTA
8	Jorge Renan da Paz	94015285433	318750369	8738693786	2019-04-02	1700	SUPERVISOR DE MECA
9	Edson Rober to Felipe Martins	64010105410	306169046	8727810963	2017-01-02	1100	MECANICO
10	Benjamin César Elias Porto	52010863496	490404327	8736045526	2020-02-02	1100	MECANICO

The 'Output' window shows the execution of the query, indicating that 10 rows were returned. The 'Table: ordenservico' window is also visible, showing columns: CODORDEMSEVICIO, ENTRADA, SAIDA, TOTAL, CODCLIENTE, PLACA, CODFUNCIONARIO, PENDENTE.

g-2) Quais são nossos Funcionários que também são clientes?

The screenshot shows the MySQL Workbench interface. The SQL Editor contains the following query:

```
SELECT * FROM CLIENTE WHERE CPF CNPJ IN (SELECT CPF CNPJ FROM FUNCIONARIO)
```

The query is executed, and the results are displayed in the Result Grid. The results show a list of clients who are also employees.

COOCLIENTE	NOME	CPF CNPJ	RG	TELEFONE	CELULAR	CODENDEREÇO	TIPOCLIENTE
5	Luís Erick Monteiro	14261490340	445235299	28266993	989907121	9	FREQUES
8	Marcelo Enrico Assis	17144517554	390515255	26325320	994077084	11	FREQUES
10	Lorenzo Thiago Nelson Santos	80518223310	419963844	39787452	996377175	12	ESPECIAL

The Output pane shows the execution details:

#	Time	Action	Message	Duration / Fetch
156	03.31.37	SELECT * FROM FUNCIONARIO LIMIT 0, 1000	10 row(s) returned	0.000 sec / 0.000 sec
157	03.31.21	SELECT * FROM CLIENTE WHERE CPF CNPJ IN (SELECT CPF CNPJ FROM FUNCIONARIO ...	3 row(s) returned	0.000 sec / 0.000 sec

g-3) Quantos funcionários ganham mais que a média de salários de todos os funcionários admitidos a partir de 2020?

Média salarial referente a admissão a partir de 2020:

The screenshot shows the MySQL Workbench interface. The SQL Editor contains the following query:

```
SELECT SUM(SALARIO)/COUNT(*) AS 'MÉDIA SALARIAL REFERENTE A ADMISSÃO EM 2020'
FROM FUNCIONARIO
WHERE DATAADMISSAO > STR_TO_DATE( "01/01/2020", "%d/%m/%Y" )
```

The query is executed, and the results are displayed in the Result Grid. The results show the average salary for employees admitted from 2020 onwards.

MÉDIA SALARIAL REFERENTE A ADMISSÃO EM 2020
1180

The Output pane shows the execution details:

#	Time	Action	Message	Duration / Fetch
169	03.46.13	SELECT SUM(SALARIO)/COUNT(*) FROM FUNCIONARIO WHERE DATAADMISSAO...	1 row(s) returned	0.000 sec / 0.000 sec
170	03.46.33	SELECT SUM(SALARIO)/COUNT(*) AS 'MÉDIA SALARIAL REFERENTE A ADMISSÃO EM ...	1 row(s) returned	0.000 sec / 0.000 sec

Quantidade de funcionários que recebem mais que a média:

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 SELECT COUNT(*) AS 'FUNCIONARIOS QUE RECEBEM MAIS QUE A MÉDIA DE SALARIO RELATIVO A ADMISSÃO EM 2020'
2 FROM FUNCIONARIO
3 WHERE
4     SALARIO > (
5         SELECT SUM(SALARIO)/COUNT(*)
6         FROM FUNCIONARIO
7         WHERE DATAADMISSAO > STR_TO_DATE( "01/01/2020", "%d/%m/%Y" )
8     );
```

The result grid shows the following output:

Result Grid
5

The output indicates that 5 employees receive more than the average salary relative to the admission in 2020.

Quem são os funcionários:

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 SELECT *
2 FROM FUNCIONARIO
3 WHERE
4     SALARIO > (
5         SELECT SUM(SALARIO)/COUNT(*)
6         FROM FUNCIONARIO
7         WHERE DATAADMISSAO > STR_TO_DATE( "01/01/2020", "%d/%m/%Y" )
8     );
```

The result grid shows the following output:

Result Grid
3
4
5
6
8

The output lists the following employees:

CODFUNCIONARIO	NOME	CPF	RG	TELEFONE	DATAADMISSAO	SALARIO	ESPECIALIDADE
3	Lorenzo Thiago Nelson Santos	80518223310	390515255	39787452	2013-01-10	1700	SUPERVISOR DE MECA
4	Marcelo da Fonseca Souza	89100047031	162565422	8739964793	2013-01-10	3000	CHEFE DE OFICINA
5	Maria Luciana Pereira	93225451436	173176884	8739940683	2020-05-13	1300	ADMINISTRAÇÃO
6	José Claudio da Silva	66901237405	485471243	8725517942	2020-05-14	1300	FATURAMENTO
8	Jorge Renan da Paz	94015285433	318750569	8738693786	2019-04-02	1700	SUPERVISOR DE MECA

g-4) Qual é a marca e modelo de veículo mais atendidos pela oficina?

Marca quem mais é atendida:

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 SELECT MARCA, COUNT(MARCA) AS QTD
2 FROM VEICULO
3 GROUP BY MARCA
4 ORDER BY QTD DESC
```

The query results are displayed in the Result Grid:

MARCA	QTD
Buggy	2
Plymouth	1
Cadillac	1
MINI	1
Ford	1
Rover	1
Engesa	1
Chrysler	1
Mahindra	1

The left sidebar shows the database schema with tables like cliente, cliente_veiculo, empresa, endereco, funcionario, item, nfservico, ordem_servico, ordemservico, servico, servico_item, usuario, and veiculo. The bottom status bar indicates "Query Completed".

Modelos dessas marcas mais atendidas:

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 SELECT * FROM VEICULO WHERE MARCA = 'Buggy'
```

The query results are displayed in the Result Grid:

PLACA	MARCA	MODELO
KJR8980	Buggy	Buggy 1.6 2-Lug.
KJQ0759	Buggy	Buggy IV e V

The left sidebar shows the database schema with tables like cliente, cliente_veiculo, empresa, endereco, funcionario, item, nfservico, ordem_servico, ordemservico, servico, servico_item, usuario, and veiculo. The bottom status bar indicates "Query Completed".

g-5) Quais são os itens de serviço dado um determinado serviço?

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 SELECT *, (SELECT DESCRICAO FROM SERVICO WHERE CODSERVICO = 4) AS 'DESCRICAO SERVICO'
2 FROM ITEM
3 WHERE CODITEM IN (SELECT CODITEM FROM SERVICO_ITEM WHERE CODSERVICO = 4)
```

The result grid shows the following data:

CODITEM	INSTRUCAO	DESCRICAO SERVICO
10	Iniciar Alinhamento e Balanceamento	Alinhamento e Balanceamento
11	Verificar se tudo está de acordo com o checklist	Alinhamento e Balanceamento
12	Finalizar Alinhamento e Balanceamento	Alinhamento e Balanceamento

The output pane shows the execution details:

#	Time	Action	Message	Duration / Fetch
188	04:01:17	SELECT *, (SELECT DESCRICAO FROM SERVICO WHERE CODSERVICO = 4) FROM ITE...	3 row(s) returned	0.000 sec / 0.000 sec
189	04:01:33	SELECT *, (SELECT DESCRICAO FROM SERVICO WHERE CODSERVICO = 4) AS DESCRICAO...	3 row(s) returned	0.000 sec / 0.000 sec

g-6) Quais são os dez Clientes para os quais as ordens de serviço demandaram maior tempo na oficina (por ordem do maior para o menor tempo).

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 SELECT (
2     SELECT NOME
3     FROM CLIENTE
4     WHERE CODCLIENTE = OS.CODCLIENTE
5 ) AS 'NOME CLIENTE',
6 DATEDIFF(SAIDA, ENTRADA) AS 'DIAS NA OFICINA'
7 FROM ORDENSERVICO OS
8 WHERE PENDENTE = 'N'
9 ORDER BY DATEDIFF(SAIDA, ENTRADA) DESC LIMIT 10
```

The result grid shows the following data:

NOME CLIENTE	DIAS NA OFICINA
Mariana de Carvalho Nunes	7
Maria da Fonseca Souza	7
Vitor Cláudio Assunção	7
Manoel Lago Julio Pereira	7
Marcelo Enrico Assis	7
Carlos Noah Leonardo de Cunha	7
Devi de Lima das Neves	3
Isaac Samuel Assis	3
Luís Erik Monteiro	3
Lorenzo Thiago Nelson Santos	3

The output pane shows the execution details:

#	Time	Action	Message	Duration / Fetch
188	04:11:36	SELECT / SELECT NOME FROM CLIENTE WHERE CODCLIENTE = OS.CODCLIENTE	10 row(s) returned	0.000 sec / 0.000 sec

g-7) Quais são os Clientes que ainda não foram atendidos (veículos na oficina)?

MySQL Workbench interface showing a query execution. The query is:

```
1 SELECT (
2   SELECT NOME
3   FROM CLIENTE
4   WHERE CODCLIENTE = OS.CODCLIENTE
5 ) AS 'NOME CLIENTE'
6 FROM ORDENSERVICO OS
7 WHERE PENDENTE = 'S'
```

The result grid shows the following data:

NOME CLIENTE
BENEDITO NATHAN DA SILVA
HENRY HECTOR DAVI SOUZA
MARCOS THIAGO JUAN CASTRO

Information panel for Table: **servico**:

Columns:

- CODSERVICO: bigint AI PK
- DESCRICAO: varchar(255)
- PRECO: float

Query Completed

g-8) Existe algum mecânico que participou de todos os consertos executados pela oficina?

Com esse resultado podemos obter que não existiu um mecânico que estava em todos os processos dos serviços.

MySQL Workbench interface showing a query execution. The query is:

```
1 SELECT DISTINCT(CODORDENSERVICO),
2 (
3   SELECT NOME
4   FROM FUNCIONARIO
5   WHERE CODFUNCIONARIO = OS.CODFUNCIONARIO
6 ) AS 'NOME MECANICO' FROM ORDENSERVICO OS
```

The result grid shows the following data:

CODORDENSERVICO	NOME MECANICO
1	Edson Roberto Felipe Martins
2	Benjamin César Elias Porto
3	Luís Erick Monteiro
4	Luís Erick Monteiro
5	Marcelo Enrico Assis
6	Marcelo Enrico Assis
7	Benjamin César Elias Porto
8	Benjamin César Elias Porto
9	Luís Erick Monteiro
10	Edson Roberto Felipe Martins
11	Luís Erick Monteiro
12	Luís Erick Monteiro
13	Luís Erick Monteiro
14	Benjamin César Elias Porto
15	Edson Roberto Felipe Martins
16	Marcelo Enrico Assis

Information panel for Table: **ordenservico**:

Columns:

- CODORDENSERVICO: bigint AI PK
- ENTRADA: datetime
- SAIDA: datetime
- TOTAL: float
- CODCLIENTE: bigint
- PLACA: varchar(10)
- CODFUNCIONARIO: bigint
- PENDENTE: boolean

Query Completed

g-9) Qual é a dívida total dos Clientes “Devedores”.

Apenas um cliente está como devedor, o mesmo está devendo a quantia a baixo referente a ordens de serviço finalizadas.

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 SELECT SUM(TOTALSERVICO) AS 'TOTAL DAS DIVIDAS'
2 FROM ORDEMSERVICO
3 WHERE CODCLIENTE IN (
4   SELECT CODCLIENTE
5   FROM CLIENTE
6   WHERE TIPOCLIENTE = 'DEVEDOR'
7 ) AND PENDENTE = 'N';
```

The query results are displayed in the Result Grid, showing a single row with the value 65.

TOTAL DAS DIVIDAS
65

The Information panel on the left shows the structure of the CLIENTE table:

Columns:	
CODCLIENTE	bigint(11)
NOME	varchar(100)
CPF/CNPJ	bigint(15)
RG	int(11)
TELEFONE	bigint(15)
CELULAR	bigint(15)
CODENDERECO	bigint(11)
TIPOCLIENTE	varchar(10)

The Action Output panel at the bottom shows the execution of the query, returning 1 row(s).

Porém este cliente também possui ordem de serviço não finalizada.

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 SELECT *
2 FROM ORDEMSERVICO
3 WHERE CODCLIENTE IN (
4   SELECT CODCLIENTE
5   FROM CLIENTE
6   WHERE TIPOCLIENTE = 'DEVEDOR'
7 );
```

The query results are displayed in the Result Grid, showing two rows of data. Red arrows point to the CODCLIENTE and PENDENTE columns for the second row.

CODORDEMSERVICO	ENTRADA	SAIDA	TOTAL	CODCLIENTE	PLACA	CODFUNCIONARIO	PENDENTE	TOTALSERVICO	VALORDESCONTO
13	2021-10-18	2021-10-18	65	13	KK3444	8	N	65	0
16	2021-11-11	0000-00-00	170	13	KK3444	3	S	170	0

The Information panel on the left shows the structure of the CLIENTE table:

Columns:	
CODCLIENTE	bigint(11)
NOME	varchar(100)
CPF/CNPJ	bigint(15)
RG	int(11)
TELEFONE	bigint(15)
CELULAR	bigint(15)
CODENDERECO	bigint(11)
TIPOCLIENTE	varchar(10)

The Action Output panel at the bottom shows the execution of the query, returning 1 row(s).

g-10) Quais são os funcionários que recebem o mesmo salário?
Quantidade de vezes que repete o salário.

The screenshot shows a SQL query in a text editor window titled "SQL File 3*". The query is as follows:

```
1 SELECT SALARIO, COUNT(SALARIO) AS QTD
2 FROM FUNCIONARIO
3 GROUP BY SALARIO
4 ORDER BY QTD DESC;
```

Below the query, the "Result Grid" is displayed, showing the results of the query. The grid has two columns: SALARIO and QTD. The results are as follows:

SALARIO	QTD
1100	5
1700	2
1300	2
3000	1

Tendo esses dados, conseguimos filtrar os funcionários que tem o salário que repete mais de uma vez.

The screenshot shows a SQL query in a text editor window titled "SQL File 3*". The query is as follows:

```
1 SELECT
2 NOME, SALARIO
3 FROM FUNCIONARIO
4 WHERE SALARIO IN (1100, 1700, 1300)
5 ORDER BY SALARIO DESC
```

Below the query, the "Result Grid" is displayed, showing the results of the query. The grid has two columns: NOME and SALARIO. The results are as follows:

NOME	SALARIO
Lorenzo Thiago Nelson Santos	1700
Jorge Renan da Paz	1700
Maria Luciana Pereira	1300
José Claudio da Silva	1300
Luís Erick Monteiro	1100
Marcelo Enrico Assis	1100
Andréia Firmino Conceição	1100
Edson Roberto Felipe Martins	1100
Benjamin César Elias Porto	1100