

DISCIPLINA: BANCO DE DADOS

**Modelo de Entidade Relacionamento - Sistema de
controle de candidaturas e gastos das legislaturas
(2022)**

DISCENTES:

JOSÉ RENAN ALVES PEREIRA
MIKAELLE DOS SANTOS OLIVEIRA

PROF^(a): FÁBIO LEITE

CAMPINA GRANDE, Outubro de 2022

[illegible]

Candidato(cpf_candidato, nome_candidato, identidade_candidato, comprovante_de_residência_candidato, certificado_de_legalidade_candidato, #número_de_inscrição título_de_eleitor*, #código_da_candidatura)

Candidatura(cod_candidatura, ano_da_candidatura,
#número_do_partido_ficha_de_filiação*)

Ficha_de_filiação(número_partido_ficha_de_filiação,
nome_partido_ficha_de_filiação)

Bem(ID_Bens, tipo_do_bem, descrição_do_bem, valor_de_mercado_do_bem, #código_da_declaração_de_bens*)

Votação(cod_Votação, resultado_da_votação,
situação_de_registro_de_candidatura_da_votação, #código_da_candidatura*)

Votação_possui_urna(#cod_votação,#cod_urna,turno,ano)

Urna(cod_urna, quantidade_de_votos_da_urna, zona_da_urna, região_da_urna,
município_da_urna, estado_da_urna)

Gasto(cod_gasto,data_gasto, descrição_do_gasto, fornecedor_do_gasto,
valor_total_do_gasto, valor_liquido_gasto, categoria_do_gasto
,#código_do_mandato*)

Doações_de_campanha(num_de_identificação_doação,
valor_doadado_doação,descrição_doação, data_doação,
hora_doação,#código_da_candidatura*)

Doador(cod_doador, #num_de_identificação_doação*)

Pessoa_física(#cod_doador,cpf_pessoa_física, nome_pessoa_física,
Cep_pessoa_física, Numero_residencia_pessoa_física, telefone_pessoa_física)

Verba_pública(#cod_doador, rúbrica_verba_pública,tipo_verba_pública,
descrição_verba_pública)

Dicionário de dados

Entidade: Candidato

Atributo	Classe	Domínio	Tamanho	Descrição
cpf	Determinante	Texto	11	Registro candidato, único.
nome	Simples	Texto	100	
identidade	Simples	Numérico		
comprovante_d e_residencia	Simples	BYTEA		Imagem do comprovante
certificado_de_l egalidade	Simples	BYTEA		Imagem do certificado
numero_de_ins cricao	Simples	Numérico		Chave estrangeira de título de eleitor, obrigatório
cod_candidatur a	Simples	Numérico		Chave estrangeira de título de candidatura, obrigatório

Entidade: Titulo_de_eleitor

Atributo	Classe	Domínio	Tamanho	Descrição
numero_de_inscricao	Determinante	Numérico		Registro do número de inscrição do título de eleitor, único
regiao	Simples	Texto	50	
zona	Simples	Numérico		

Entidade: Candidatura

Atributo	Classe	Domínio	Tamanho	Descrição
cod_candidatura	Determinante	Numérico		Registro do código de candidatura, único
ano	Simples	Numérico		Ano de candidatura
num_partido	Simples	Numérico		Chave estrangeira de ficha de filiação, obrigatório

Entidade: Mandato

Atributo	Classe	Domínio	Tamanho	Descrição
cod_mandato	Determinante	Numérico		Registro do código do mandato, único

Entidade: Ficha_de_filiação

Atributo	Classe	Domínio	Tamanho	Descrição
cod_filiacao	Determinante	Numérico		Registro do código da ficha de filiação, único
num_partido	Simples	Numérico		
nome_partido	Simples	Texto	50	

Entidade: Declaração_de_bens

Atributo	Classe	Domínio	Tamanho	Descrição
cod_declaracao	Determinante	Numérico		Registro da declaração, único

Entidade: Bem

Atributo	Classe	Domínio	Tamanho	Descrição
id_bem	Determinante	Numérico		Registro do bem, único
descricao	Simples	Texto	60	Descrição de como o bem em questão se encontra
valor	Simples	Texto		Preço do bem
tipo	Simples	Numérico	15	Imóveis, veículos, licença e concessão especial, jóias, obras de arte etc.
cod_declaracao	Simples	Numérico		Chave estrangeira de declaração de bens, obrigatório

Entidade: Votação

Atributo	Classe	Domínio	Tamanho	Descrição
cod_votacao	Determinante	Numérico		Registro de votação, único
resultado	Simples	Texto	15	Eleito, indeferido com recurso, não eleito, suplente, substituto, renúncia ou falecimento, registro negado
situacao	Simples	Texto	15	Apto, inapto, cadastrado

Entidade: Votação_possui_urna

Atributo	Classe	Domínio	Tamanho	Descrição
cod_votacao	Determinante	Numérico		Chave primária estrangeira, vinda de votação, único

cod_urna	Determinante	Numérico		Chave primária estrangeira, vinda de urna, único
ano	Simples	Numérico		
turno	Simples	Numérico		

Entidade: Urna

Atributo	Classe	Domínio	Tamanho	Descrição
cod_urna	Determinante	Numérico		Registro da urna, único
votos	Simples	Numérico		
estado	Simples	Texto	20	
regiao	Simples	Texto	20	
municipio	Simples	Texto	50	
zona	Simples	Numérico		

Entidade: Gasto

Atributo	Classe	Domínio	Tamanho	Descrição
cod_gasto	Determinante	Numérico		Registro do gasto, único
descricao	Simples	Texto	60	
fornecedor	Simples	Texto	30	
data_gasto	Simples	Texto	8	Formato: dd-mm-aaaa:
valor_total	Simples	Numérico		
valor_liquido	Simples	Numérico		
categoria	Simples	Texto	20	

Entidade: Doação_de_campanha

Atributo	Classe	Domínio	Tamanho	Descrição
id_doacao	Determinante	Numérico		Registro da doação, único

descricao	Simples	Texto	60	
valor_doador	Simples	Numérico		
data_doacao	Simples	Texto	8	Formato: dd-mm-aaaa:
hora	Simples	Texto	6	Formato: ## : ## PM

Entidade: Doador

Atributo	Classe	Domínio	Tamanho	Descrição
cod_doador	Determinante	Numérico		Registro de doador, único

Entidade: Pessoa_física

Atributo	Classe	Domínio	Tamanho	Descrição
cod_doador	Determinante	Numérico		Chave estrangeira de doador, único
cpf	Simples	Texto	11	único, obrigatório
nome	Simples	Texto	100	
endereco	Simples	Texto	100	
telefone	Simples	Texto	13	

Entidade: Verba_pública

Atributo	Classe	Domínio	Tamanho	Descrição
cod_doador	Determinante	Numérico		Chave estrangeira de doador, único
descricao	Simples	Texto	60	
tipo	Simples	Texto	25	Repasse do fundo partidário, doações do próprio partido, etc.
rubrica	Simples	Texto	25	

Script de criação do sql

--- TABELA REFERENTE AO TÍTULO DE ELEITOR

```
CREATE TABLE IF NOT EXISTS TITULO_DE_ELEITOR (  
    numero_de_inscricao      INTEGER          NOT NULL,  
    regiao                   VARCHAR(50),  
    zona                     INTEGER,  
    cod_candidato            INTEGER,  
    PRIMARY KEY (numero_de_inscricao),  
    UNIQUE (cod_candidato));
```

--- TABELA REFERENTE A FICHA_DE_FILIACAO

```
CREATE TABLE IF NOT EXISTS FICHA_DE_FILIACAO(  
    cod_filiacao              INTEGER          NOT NULL,  
    num_partido               INTEGER unique   NOT NULL,  
    nome_partido              VARCHAR(50),  
    PRIMARY KEY (cod_filiacao));
```

--- TABELA REFERENTE A DECLARACAO_DE_BENS

```
CREATE TABLE IF NOT EXISTS DECLARACAO_DE_BENS (  
    cod_declaracao            INTEGER          NOT NULL,  
    PRIMARY KEY (cod_declaracao));
```

--- TABELA REFERENTE AOS BENS

```
CREATE TABLE IF NOT EXISTS BEM (  
    id_bem                    INTEGER          NOT NULL,  
    descricao                 VARCHAR(60),  
    valor                     DOUBLE PRECISION,  
    tipo                      VARCHAR(15),  
    cod_declaracao            INTEGER          NOT NULL,  
    PRIMARY KEY (id_bem),  
    FOREIGN KEY (cod_declaracao) REFERENCES  
DECLARACAO_DE_BENS(cod_declaracao));
```

--- TABELA REFERENTE AO CANDIDATO

```
CREATE TABLE IF NOT EXISTS CANDIDATO (  
    cpf                       VARCHAR(11) UNIQUE NOT NULL,  
    nome                      VARCHAR(100),  
    identidade                 INTEGER          ,  
    comprovante_de_residencia BYTEA ,  
    certificado_de_legalidade BYTEA ,  
    numero_de_inscricao        INTEGER          NOT NULL,  
    cod_candidatura            INTEGER          NOT NULL,  
    PRIMARY KEY (cpf),  
    FOREIGN KEY (numero_de_inscricao) REFERENCES titulo_de_eleitor  
(numero_de_inscricao),  
    UNIQUE (certificado_de_legalidade));
```


--- TABELA REFERENTE A CANDIDATURA

```
CREATE TABLE IF NOT EXISTS CANDIDATURA (  
  cod_candidatura          INTEGER          NOT NULL,  
  ano                      INTEGER,  
  num_partido              INTEGER          NOT NULL,  
  cpf                      VARCHAR(11) UNIQUE NOT NULL,  
  cod_doacao               INTEGER          NOT NULL,  
  PRIMARY KEY (cod_candidatura),  
  FOREIGN KEY (num_partido) REFERENCES ficha_de_filiacao(num_partido),  
  FOREIGN KEY (cpf) REFERENCES Candidato(cpf));
```

--- TABELA REFERENTE A DOACAO_DE_CAMPANHA

```
CREATE TABLE IF NOT EXISTS DOACAO_DE_CAMPANHA (  
  id_doacao                INTEGER          NOT NULL,  
  descricao                VARCHAR(60) ,  
  valor_doador             DOUBLE PRECISION,  
  data_doacao              VARCHAR(8),  
  hora                     VARCHAR(6),  
  cod_candidatura          INTEGER          NOT NULL  
  UNIQUE,  
  PRIMARY KEY (id_doacao),  
  FOREIGN KEY (cod_candidatura) REFERENCES  
  Candidatura(cod_candidatura));
```

--- TABELA REFERENTE AO MANDATO

```
CREATE TABLE IF NOT EXISTS MANDATO (  
  cod_mandato              INTEGER          NOT NULL,  
  cod_candidatura          INTEGER          NOT NULL  
  UNIQUE,  
  PRIMARY KEY (cod_mandato),  
  FOREIGN KEY (cod_candidatura) REFERENCES  
  Candidatura(cod_candidatura));
```

--- TABELA REFERENTE AO DOADOR

```
CREATE TABLE IF NOT EXISTS DOADOR (  
  cod_doador               INTEGER          NOT NULL,  
  PRIMARY KEY (cod_doador));
```

--- TABELA REFERENTE A PESSOA_FISICA

```
CREATE TABLE IF NOT EXISTS PESSOA_FISICA (  
  cod_doador               INTEGER          NOT NULL,  
  cpf                      VARCHAR(11) UNIQUE NOT NULL,  
  nome                     VARCHAR(100),  
  endereco                 VARCHAR(100),  
  telefone                 VARCHAR(13) ,  
  PRIMARY KEY (cod_doador, cpf),
```

```

FOREIGN KEY (cod_doador) REFERENCES DOADOR (cod_doador));

--- TABELA REFERENTE A VERBA_PUBLICA
CREATE TABLE IF NOT EXISTS VERBA_PUBLICA (
    cod_doador                INTEGER                NOT NULL,
    descricao                 VARCHAR(60),
    tipo                      VARCHAR(25),
    rubrica                   VARCHAR(25),
    PRIMARY KEY (cod_doador),
    FOREIGN KEY (cod_doador) REFERENCES DOADOR (cod_doador));

--- TABELA REFERENTE AO GASTO
CREATE TABLE IF NOT EXISTS GASTO (
    cod_gasto                 INTEGER                NOT NULL,
    descricao                 VARCHAR(60),
    fornecedor                VARCHAR(30),
    data_gasto                VARCHAR(8),
    valor_total                DOUBLE PRECISION,
    valor_liquido              DOUBLE PRECISION,
    categoria                  VARCHAR(20),
    cod_mandato                INTEGER NOT NULL,
    PRIMARY KEY (cod_gasto),
    FOREIGN KEY (cod_mandato) REFERENCES Mandato(cod_mandato));

--- TABELA REFERENTE A URNA
CREATE TABLE IF NOT EXISTS URNA (
    cod_urna                  INTEGER                NOT NULL,
    votos                     INTEGER,
    estado                     VARCHAR(20),
    regiao                     VARCHAR(20),
    municipio                  VARCHAR(50),
    zona                       INTEGER,
    PRIMARY KEY (cod_urna),
    UNIQUE (municipio));

--- TABELA REFERENTE A VOTACAO
CREATE TABLE IF NOT EXISTS VOTACAO (
    cod_votacao               INTEGER                NOT NULL,
    resultado                  VARCHAR(15)           NOT NULL,
    situacao                   VARCHAR(15)           NOT NULL,
    cod_candidatura            INTEGER                NOT NULL UNIQUE,
    PRIMARY KEY (cod_votacao),
    FOREIGN KEY (cod_candidatura) REFERENCES
Candidatura(cod_candidatura));

--- TABELA REFERENTE A VOTACAO_POSSUI_URNA

```

```

CREATE TABLE IF NOT EXISTS VOTACAO_POSSUI_URNA (
    cod_votacao          INTEGER          NOT NULL,
    cod_urna              INTEGER          NOT NULL,
    ano                  INTEGER,
    turno                 INTEGER,
    PRIMARY KEY (cod_votacao, cod_urna),
    FOREIGN KEY (cod_votacao) REFERENCES VOTACAO (cod_votacao),
    FOREIGN KEY (cod_urna) REFERENCES URNA (cod_urna));

```

Scripts de inserção do sql

--- INSERINDO OS DADOS NA TABELA DE TITULO_DE_ELEITOR

```

INSERT INTO TITULO_DE_ELEITOR
(numero_de_inscricao, regiao, zona)
VALUES
(268741, 'Pocinhos', 01),
(175924, NULL, 02),
(374904, 'Campina Grande', 03),
(782905, 'João Pessoa', 07),
(781200, 'Guarabira', 05),
(094837, NULL, 023),
(760391, NULL, 012),
(174320, 'Soledade', 085),
(580296, 'Campina Grande', 325),
(892736, 'Areia', 028),
(374905, 'Montadas', 072);

```

--- INSERINDO OS DADOS NA TABELA DE FICHA_DE_FILIACAO

```

INSERT INTO FICHA_DE_FILIACAO
(cod_filiacao, num_partido, nome_partido)
VALUES
(294138, 087, 'PDT'),
(032054, 014, 'PT'),
(780940, 01, 'PL'),
(030972, 03, 'PMDB'),
(204795, 06, 'PSOL'),
(587496, 05, 'CDU');

```

--- INSERINDO OS DADOS NA TABELA DE DECLARACAO_DE_BENS

```

INSERT INTO DECLARACAO_DE_BENS
(cod_declaracao)

```

VALUES

(3682),
(8194),
(2790),
(0379),
(3719),
(7920),
(9348),
(9372),
(5872),
(7930);

--- INSERINDO OS DADOS NA TABELA DE BEM

INSERT INTO BEM

(id_bem, descricao, valor, tipo, cod_declaracao)

VALUES

(7297, NULL, 8, 'Caneta', 379),
(0374, 'Carro familiar', 95000, 'Carro', 3682),
(0495, NULL, 12000, 'Moto', 2790),
(3862, NULL, 56000, 'Casa', 3719),
(7925, 'Casa familiar', 120000, 'Casa', 5872),
(5216, NULL, 460, 'Bicicleta', 7920),
(4360, 'Telefona familiar', 560, 'Telefone fixo', 7930),
(5964, NULL, 450, NULL, 8194);

--- INSERINDO OS DADOS NA TABELA DE CANDIDATO

INSERT INTO CANDIDATO

(cpf, nome, identidade, comprovante_de_residencia,
certificado_de_legalidade, numero_de_inscricao, cod_candidatura)

VALUES

('01259832157', 'Renan', 297585, NULL, NULL, 760391, 298068),
('58921765402', 'Italo', NULL, NULL, NULL, 175924, 298003),
('02798239467', 'Fábio', 282470, NULL, NULL, 892736, 158285),
('59212350921', 'Juliana', 892175, NULL, NULL, 781200, 97160),
('01259872157', 'Mikaele', 291085, NULL, NULL, 374904, 298068),
('14559872100', 'Ramon', 1470085, NULL, NULL, 374904, 258415),
('04020872577', 'Junior', 5819360, NULL, NULL, 374904, 854796),
('85603159205', 'Caroline', 0325601, NULL, NULL, 580296, 587195),
('14785236931', 'Caio', 505111, NULL, NULL, 374905, 968574),
('00165435744', 'Natalia', 130006, NULL, NULL, 892736, 580691),
('95124876189', 'Maria', 240600, NULL, NULL, 94837, 298067);

--- INSERINDO OS DADOS NA TABELA DE CANDIDATURA

INSERT INTO CANDIDATURA

(cod_candidatura, ano, num_partido, cpf, cod_doacao)

VALUES

```
(298003, 2010, 01, '58921765402', 5972),
(97160, 2018, 03, '59212350921', 5209),
(158285, 2016, 014, '02798239467', 0579),
(298068, 2022, 06, '01259872157', 7892),
(258415, 2010, 01, '14559872100', 5209),
(854796, 2018, 05, '04020872577', 0579),
(968574, 2002, 05, '14785236931', 3981),
(580691, 2004, 05, '00165435744', 7892),
(298067, 2006, 05, '95124876189', 0684),
(587195, 2018, 05, '85603159205', 0684);
```

--- INSERINDO OS DADOS NA TABELA DE DOACAO_DE_CAMPANHA

```
INSERT INTO DOACAO_DE_CAMPANHA
```

```
(id_doacao, descricao, valor_doador, data_doacao, hora, cod_candidatura)
VALUES
```

```
(3981, NULL, 36.000, NULL, NULL, 97160),
(0684, 'Doação para propaganda', 300, NULL, '12:12', 158285),
(2418, NULL, 85.00, '12022010', NULL, 298068),
(0579, 'Locomoção', 60.000, '20102006', '16:57', 258415),
(5972, NULL, 21.00, NULL, NULL, 968574),
(0289, NULL, NULL, NULL, NULL, 854796),
(8957, 'Transporte', 40.000, '20112022', '22:32', 298003),
(7892, 'Doação de um avião', 100.000, '25062006', '08:45', 587195),
(5209, 'Divulgação nas ruas', 20.000, NULL, '14:47', 298067);
```

--- INSERINDO OS DADOS NA TABELA DE DOADOR

```
INSERT INTO DOADOR
```

```
(cod_doador)
```

```
VALUES
```

```
(749163),
(792480),
(267409),
(097262),
(549270);
```

--- INSERINDO OS DADOS NA TABELA DE PESSOA_FISICA

```
INSERT INTO PESSOA_FISICA
```

```
(cod_doador, cpf, nome, endereco, telefone)
```

```
VALUES
```

```
(749163, '19856702196', 'Gustavo', NULL, '25394821'),
(792480, '48298726513', NULL, NULL, NULL),
(267409, '57436598752', 'Larissa', 'Bairro da Prata, Campina Grande',
'58412563'),
(097262, '25418567404', 'Emanuel', 'Bairro do Cruzeiro, Pocinhos',
'25039745'),
(549270, '72942801698', 'Fanuel', NULL, '58694125');
```

--- INSERINDO OS DADOS NA TABELA DE VERBA_PUBLICA

INSERT INTO VERBA_PUBLICA

(cod_doador, descricao, tipo, rubrica)

VALUES

(749163, 'Propaganda eleitoral', 'Fundo partidário', NULL),
(792480, 'Propaganda eleitoral', 'Fundo partidário', 7818),
(267409, 'Propaganda eleitoral', 'Fundo partidário', 7825),
(097262, 'Propaganda eleitoral', 'Fundo partidário', 0287);

--- INSERINDO OS DADOS NA TABELA DE MANDATO

INSERT INTO MANDATO

(cod_mandato, cod_candidatura)

VALUES

(1, 298003),
(2, 097160),
(3, 158285),
(4, 298068),
(5, 258415),
(6, 854796),
(7, 968574),
(8, 580691),
(9, 298067);

--- INSERINDO OS DADOS NA TABELA DE GASTO

INSERT INTO GASTO

(cod_gasto, descricao, fornecedor, data_gasto, valor_total,
valor_liquido, categoria, cod_mandato)

VALUES

(748294, 'Propaganda', 'Verba pública', NULL, NULL, NULL, NULL, 1),
(087290, NULL, 'Verba pública', '10022006', 520000, 580000, NULL, 2),
(684292, 'Propaganda', 'Verba pública', NULL, 9600000, NULL,
'Campanha', 3),
(782902, 'Propaganda', 'Doação privada', '25122018', NULL, 120000,
NULL, 4),
(524308, 'Outdoors', 'Doação privada', NULL, 20000, 5600000, NULL, 9),
(122590, NULL, 'Fundo eleitoral', NULL, 5600, 5600, NULL, 5),
(201908, 'Outdoors', 'Fundo eleitoral', NULL, 6900, 6900, NULL, 6),
(194030, 'Carreata', 'Fundo eleitoral', NULL, 98000, 98000, NULL, 7),
(861377, 'Carreata', 'Doação privada', NULL, 77000, 77000, NULL, 8);

--- INSERINDO OS DADOS NA TABELA DE URNA

INSERT INTO URNA

(cod_urna, votos, estado, regioao, municipio, zona)

VALUES

(01, 123, 'Paraiba', NULL, 'Boa Vista', 06),

```
(02, 201, 'São Paulo', 'Sudeste', 'São Paulo', 047),
(03, 265, 'Alagoas', NULL, NULL, 069),
(04, 167, 'Paraíba', 'Nordeste', 'Maceió', 0123),
(05, 197, 'Paraíba', NULL, 'Pocinhos', 010),
(06, 230, 'São Paulo', 'Sudeste', 'Campinas', 052),
(07, 214, 'Alagoas', 'Nordeste', 'Penedo', 0254),
(08, 306, 'Alagoas', 'Nordeste', 'São Miguel dos Milagres', 0231),
(09, 185, 'Alagoas', 'Nordeste', 'Porto das Pedras', 0201),
(10, 600, 'Paraíba', 'Nordeste', 'Campina Grande', 0221);
```

--- INSERINDO OS DADOS NA TABELA DE VOTACAO

```
INSERT INTO VOTACAO
```

```
(cod_votacao, resultado, situacao, cod_candidatura)
```

```
VALUES
```

```
(482798, 'Eleito', 'Apto', 298003),
(829429, 'suplente', 'inapto', 097160),
(048217, 'Não eleito', 'Apto', 158285),
(859182, 'Eleito', 'Apto', 587195),
(653224, 'Eleito', 'Apto', 854796),
(730125, 'Não eleito', 'Inapto', 968574),
(410630, 'Substituto', 'Apto', 580691),
(400106, 'Eleito', 'Apto', 298067);
```

--- INSERINDO OS DADOS NA TABELA DE VOTACAO_POSSUI_URNA

```
INSERT INTO VOTACAO_POSSUI_URNA
```

```
(cod_votacao, cod_urna, ano, turno)
```

```
VALUES
```

```
(48217, 1, 2006, 1),
(482798, 6, 2010, 1),
(829429, 7, 2022, 2),
(859182, 10, 2018, 2),
(730125, 02, 2006, 1),
(410630, 04, 2010, 1),
(400106, 08, 2022, 2);
```

Script das consultas do sql

--- A)

--- Liste os candidatos pelo nome e CPF que obtiveram votos na cidade de "Campina Grande" e que gastaram mais de R\$ 5.000,00 na campanha;

```
select DISTINCT c.nome as "Nome", c.cpf as "CPF"
from candidato c, votacao v, urna u, titulo_de_eleitor te, gasto ga,
mandato m
```

```
where c.numero_de_inscricao = te.numero_de_inscricao and u.municipio =  
te.regiao AND c.cod_candidatura = m.cod_candidatura  
and m.cod_mandato = ga.cod_mandato  
and ga.valor_total > 5000
```

--- B)

--- Liste os candidatos pelo nome e CPF e código do partido, total declarado, total recebido em doação e total gasto em campanha que possuíram mais de 500 votos na eleição de 2018 na cidade de Campina Grande;

```
select DISTINCT c.nome as "Nome", c.cpf as "CPF", fl.num_partido AS  
"Número do partido", SUM(dc.valor_doado) as "Total doado",  
sum(g.valor_total) as "Total recebido"  
from votacao v inner join votacao_possui_urna vpu on (v.cod_votacao =  
vpu.cod_votacao) INNER join urna u on (vpu.cod_urna = u.cod_urna),  
candidato c inner join candidatura cdt on (c.cod_candidatura =  
cdt.cod_candidatura), ficha_de_filiacao fl, doacao_de_campanha dc, gasto  
g  
where v.cod_votacao = vpu.cod_votacao and u.cod_urna = vpu.cod_urna and  
u.municipio like 'Campina Grande'  
and vpu.ano = 2018 and u.votos > 500 and v.cod_candidatura =  
cdt.cod_candidatura and fl.num_partido = cdt.num_partido  
group by c.nome, c.cpf, fl.num_partido
```

--- C)

--- Selecione o total de doação partidária recebido pelos candidatos do partido 'CDU'.

```
select f.nome_partido as "Partido", SUM(d.valor_doado) as "Total doado"  
from Ficha_de_filiacao as f, candidatura as ctra, Doacao_de_campanha as  
d, Candidato as c  
where f.nome_partido = 'CDU' and f.num_partido = ctra.num_partido  
and ctra.cod_candidatura = d.cod_candidatura and ctra.cpf = c.cpf  
GROUP BY f.nome_partido;
```

--- D)

--- Apresente uma relação dos 3 partidos que receberam maior número de verbas partidárias

```
SELECT f.nome_partido as "Partido", f.num_partido as "Número do partido"  
, SUM(d.valor_doado) as "Verbas"  
FROM Doacao_de_campanha as d, Ficha_de_filiacao as f, Candidatura as  
ctra  
WHERE d.cod_candidatura = ctra.cod_candidatura
```



```
        and ctra.num_partido = f.num_partido
GROUP BY f.nome_partido, f.num_partido ORDER BY SUM(d.valor_doado) DESC
LIMIT 3;
```

--- E)

--- Apresente uma lista de todas as cidades e seus respectivos quantitativos de votos nos candidatos do partido 'CDU'. Ordene pela quantidade de votos válidos.

```
select f.nome_partido as "Nome do partido", sum(u.votos) as "Votos",
u.municipio as "Cidade"
from urna as u, ficha_de_filiacao as f, candidatura as ctra, votacao as
v, votacao_possui_urna as vpu, candidato as c
where f.nome_partido = 'CDU' and ctra.num_partido = f.num_partido
        and v.cod_candidatura = ctra.cod_candidatura and v.cod_votacao =
vpu.cod_votacao
        and vpu.cod_urna = u.cod_urna AND ctra.cpf = c.cpf
group by f.nome_partido, u.municipio order by sum(u.votos) desc;
```