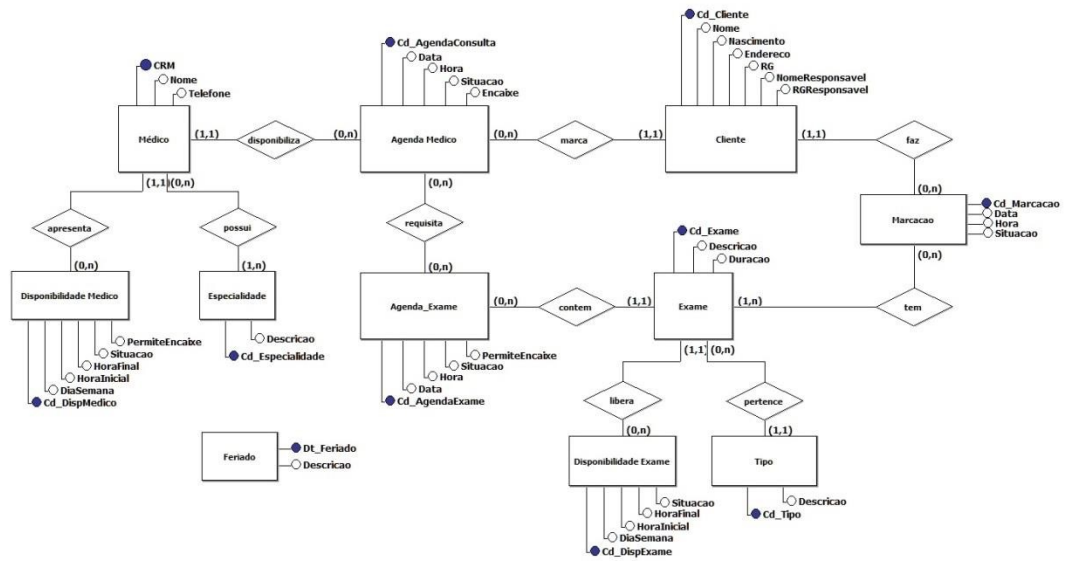


PROJETO DE BANCO DE DADOS RELACIONAL

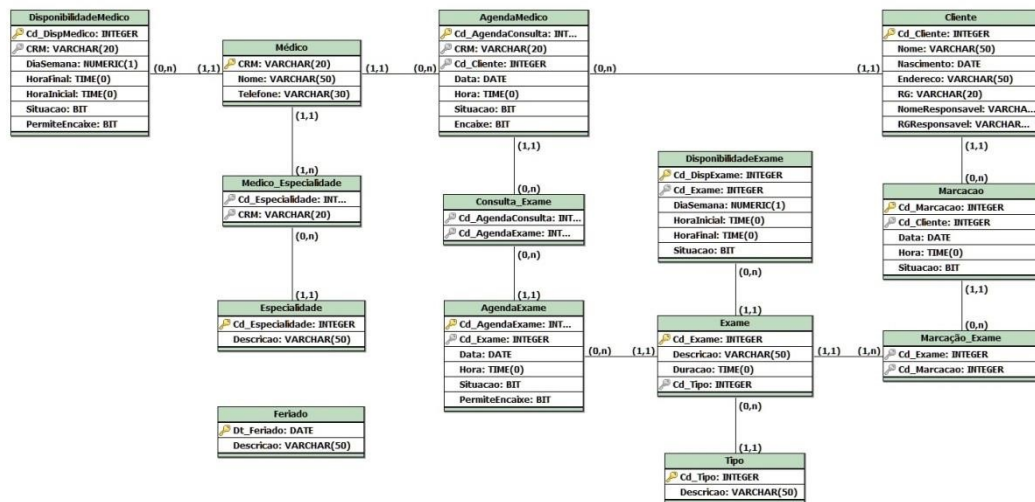
Sistema de Clínica Popular - SCP

Alunos:

Modelo Conceitual



Modelo Lógico



CRIAÇÃO DAS TABELAS

Feriado

CAMPO	TIPO	CHAVE	NÃO NULO
Dt_Feriado	DATE	PK	X
Descricao	VARCHAR(50)		X

Especialidade

CAMPO	TIPO	CHAVE	NÃO NULO
Cd_Especialidade	INTEGER IDENTITY	PK	X
Descricao	VARCHAR(50)		X

Medico

CAMPO	TIPO	CHAVE	NÃO NULO
CRM	VARCHAR(20)	PK	X
Nome	VARCHAR(50)		X
Telefone	VARCHAR(30)		X

Medico_Especialidade

CAMPO	TIPO	CHAVE	NÃO NULO
CRM	VARCHAR(20)	PK	X
Cd_Especialidade	INTEGER	FK	X

DisponibilidadeMedico

CAMPO	TIPO	CHAVE	NÃO NULO
Cd_DispMedico	INTEGER IDENTITY	PK	X
CRM	VARCHAR(20)	FK	X
DiaSemana	NUMERIC(1)		X
Horainicial	TIME(0)		X
HoraFinal	TIME(0)		X
Situacao	BIT		X
PermiteEncaixe	BIT		X

Cliente

CAMPO	TIPO	CHAVE	NÃO NULO
Cd_Cliente	INTEGER IDENTITY	PK	X
Nome	VARCHAR(50)		X
Nascimento	DATE		X
Endereco	VARCHAR(50)		X
RG	VARCHAR(20)		X
NomeResponsavel	VARCHAR(50)		X
RGResponsavel	VARCHAR(20)		X

AgendaMedico

CAMPO	TIPO	CHAVE	NÃO NULO
Cd_AgendaMedico	INTEGER IDENTITY	PK	X
CRM	VARCHAR(20)	FK	X
Cd_Cliente	INTEGER	FK	
Data	DATE		X
Hora	TIME(0)		X
Situacao	BIT		X
Encaixe	BIT		X

Tipo

CAMPO	TIPO	CHAVE	NÃO NULO
Cd_Tipo	INTEGER IDENTITY	PK	X
Descricao	VARCHAR(50)		X

Exame

CAMPO	TIPO	CHAVE	NÃO NULO
Cd_Exame	INTEGER IDENTITY	PK	X
Descricao	VARCHAR(50)		X
Duracao	TIME(0)		X
Cd_Tipo	INTEGER	FK	X

DisponibilidadeExame

CAMPO	TIPO	CHAVE	NÃO NULO
Cd_DispExame	INTEGER IDENTITY	PK	X
Cd_Exame	INTEGER	FK	X
DiaSemana	NUMERIC(1)		X
HoraInicial	TIME(0)		X
HoraFinal	TIME(0)		
Situacao	BIT		

AgendaExame

CAMPO	TIPO	CHAVE	NÃO NULO
Cd_AgendaExame	INTEGER IDENTITY	PK	X
Cd_Exame	INTEGER	FK	X
Data	DATE		X
Hora	TIME(0)		X
Situacao	BIT		X
PermiteEncaixe	BIT		X

Consulta_Exame

CAMPO	TIPO	CHAVE	NÃO NULO
Cd_AgendaMedico	INTEGER	FK	X
Cd_AgendaExame	INTEGER	FK	X

Marcacao

CAMPO	TIPO	CHAVE	NÃO NULO
Cd_Marcacao	INTEGER IDENTITY	PK	X
Cd_Cliente	INTEGER	FK	X
Data	DATE		X
Hora	TIME(0)		X
Situacao	BIT		X

Marcacao_Exame

CAMPO	TIPO	CHAVE	NÃO NULO
Cd_Marcacao	INTEGER	FK	X
Cd_Exame	INTEGER	FK	X

Scripts de Criação das Tabelas

SET LANGUAGE portuguese;

If DB_ID('Clinica') IS NOT NULL

BEGIN

USE Master

DROP DATABASE Clinica

END

CREATE DATABASE Clinica

GO

USE Clinica

CREATE TABLE Feriado (Dt_Feriado DATE PRIMARY KEY,
Descricao VARCHAR(50) NOT NULL)

CREATE TABLE Especialidade (Cd_Especialidade INTEGER IDENTITY PRIMARY KEY,
Descricao VARCHAR(50) NOT NULL)

CREATE TABLE Medico (CRM VARCHAR(20) PRIMARY KEY,
Nome VARCHAR(50) NOT NULL,
Telefone VARCHAR(30) NOT NULL)

CREATE TABLE Medico_Especialidade (CRM VARCHAR(20) REFERENCES Medico(CRM)
NOT NULL,
Cd_Especialidade INTEGER REFERENCES Especialidade(Cd_Especialidade)
NOT NULL)

CREATE TABLE DisponibilidadeMedico (Cd_DispMedico INTEGER IDENTITY PRIMARY KEY,
CRM VARCHAR(20) REFERENCES Medico(CRM) NOT NULL,
DiaSemana NUMERIC(1) NOT NULL,
HoraInicial TIME(0) NOT NULL,
HoraFinal TIME(0) NOT NULL,
Situacao BIT NOT NULL,
PermiteEncaixe BIT NOT NULL)

```

CREATE TABLE Cliente      (Cd_Cliente      INTEGER IDENTITY PRIMARY KEY,
                             Nome            VARCHAR(50)      NOT NULL,
                             Nascimento       DATE            NOT NULL,
                             Endereco         VARCHAR(50)      NOT NULL,
                             RG               VARCHAR(20)      NOT NULL,
                             NomeResponsavel  VARCHAR(50)      NOT NULL,
                             RGResponsavel    VARCHAR(20)      NOT NULL)

```

```

CREATE TABLE AgendaMedico  (Cd_AgendaMedico INTEGER IDENTITY PRIMARY KEY,
                             CRM              VARCHAR(20)      REFERENCES Medico(CRM) NOT NULL,
                             Cd_Cliente       INTEGER          REFERENCES Cliente(Cd_Cliente),
                             Data             DATE            NOT NULL,
                             Hora             TIME(0)         NOT NULL,
                             Situacao         BIT             NOT NULL,
                             Encaixe          BIT             NOT NULL)

```

```

CREATE TABLE AgendaControle (Nr_Movimentacao INTEGER IDENTITY PRIMARY KEY,
                               Cd_AgendaMedico INTEGER          ,
                               CRM              VARCHAR(20)      ,
                               Cd_Cliente       INTEGER          ,
                               Data             DATE            ,
                               Hora             TIME(0)         ,
                               Situacao         BIT             ,
                               Encaixe          BIT             ,
                               DataMovimentacao DATE            ,
                               HoraMovimentacao TIME(0)         )

```

```

CREATE TABLE Tipo          (Cd_Tipo          INTEGER IDENTITY PRIMARY KEY,
                             Descricao        VARCHAR(50)      NOT NULL)

```

```

CREATE TABLE Exame         (Cd_Exame        INTEGER IDENTITY PRIMARY KEY,
                             Descricao        VARCHAR(50)      NOT NULL,
                             Duracao          TIME(0)         NOT NULL,

```

Cd_Tipo *INTEGER* *REFERENCES Tipo(Cd_Tipo)* *NOT NULL*)

CREATE TABLE DisponibilidadeExame (*Cd_DispExame* *INTEGER IDENTITY* *PRIMARY KEY*,

Cd_Exame *INTEGER* *REFERENCES Exame(Cd_Exame)* *NOT NULL*,

DiaSemana *NUMERIC(1)* *NOT NULL*,

HoraInicial *TIME(0)* *NOT NULL*,

HoraFinal *TIME(0)* *NOT NULL*,

Situacao *BIT* *NOT NULL*)

CREATE TABLE AgendaExame (*Cd_AgendaExame* *INTEGER IDENTITY* *PRIMARY KEY*,

Cd_Exame *INTEGER* *REFERENCES Exame(Cd_Exame)* *NOT NULL*,

Data *DATE* *NOT NULL*,

Hora *TIME(0)* *NOT NULL*,

Situacao *BIT* *NOT NULL*,

PermiteEncaixe *BIT* *NOT NULL*)

CREATE TABLE Consulta_Exame (*Cd_AgendaMedico* *INTEGER* *REFERENCES*
AgendaMedico(Cd_AgendaMedico) *NOT NULL*,

Cd_AgendaExame *INTEGER* *REFERENCES AgendaExame(Cd_AgendaExame)*
NOT NULL)

CREATE TABLE Marcacao (*Cd_Marcacao* *INTEGER IDENTITY* *PRIMARY KEY*,

Cd_Cliente *INTEGER* *REFERENCES Cliente(Cd_Cliente)* *NOT NULL*,

Data *DATE* *NOT NULL*,

Hora *TIME(0)* *NOT NULL*,

Situacao *BIT* *NOT NULL*)

CREATE TABLE Marcacao_Exame (*Cd_Marcacao* *INTEGER* *REFERENCES*
Marcacao(Cd_Marcacao) *NOT NULL*,

Cd_Exame *INTEGER* *REFERENCES Exame(Cd_Exame)* *NOT NULL*)

INCLUSÃO DE DADOS**Especialidade**

Descricao
Clínico Geral
Cardiologista
Pediatra
Ortopedista
Neurologista

Medico

CRM	Nome	Telefone
100001	José Pedro Buarque	3012-4212
100002	Maria Luiza Pontes	3123-5323
100003	Silvana Maria Fraga	3234-6434
100004	Marisilva Maria Santos	3345-7545
100005	Maria José da Silva	3456-8656
100006	Carlos André Nogueira	3567-9767
100007	Ana Cristina Vargas	3678-0878
100008	Manoel Carlos da Silva	3789-1989
100009	João Batista Santos	3890-1090
100010	Cristiana Silva Santos	3901-1101

Medico_Especialidade

CRM	Cd_Especialidade
100001	1
100001	2
100002	3
100002	4
100003	2
100003	3
100003	5
100004	1
100004	2
100005	4
100005	5
100006	2
100006	3
100007	1
100008	3

100008	4
100009	5
100010	2
100010	4

Disponibilidade_Medico

CRM	DiaSemana	Horainicial	HoraFinal	Situacao	PermiteEncaixe
100001	2	08:00	12:00	1	1
100001	4	08:00	12:00	1	0
100001	6	08:00	12:00	1	1
100002	2	13:00	17:00	1	0
100002	4	13:00	17:00	1	1
100002	6	13:00	17:00	1	0
100003	3	08:00	12:00	1	1
100003	5	08:00	12:00	1	0
100003	7	08:00	12:00	0	0
100004	3	13:00	17:00	1	0
100004	5	13:00	17:00	1	0
100004	7	13:00	17:00	1	0
100005	2	08:00	12:00	1	1
100005	4	08:00	12:00	1	1
100005	6	08:00	12:00	1	1
100006	2	13:00	17:00	1	0
100006	4	13:00	17:00	1	1
100006	6	13:00	17:00	1	0
100007	3	08:00	12:00	1	1
100007	5	08:00	12:00	1	1
100008	3	13:00	17:00	1	0
100008	5	13:00	17:00	1	0
100009	4	08:00	12:00	1	0
100009	6	08:00	12:00	1	1
100010	5	13:00	17:00	1	1
100010	7	13:00	17:00	1	1

Feriados

Dt_Feriado	Descricao
01/01/2013	Confraternização Universal
12/02/2013	Carnaval
29/03/2013	Paixão de Cristo
21/04/2013	Tiradentes
01/05/2013	Dia do Trabalho
30/06/2013	Corpus Christi
07/09/2013	Independência do Brasil
12/10/2013	Nossa Sra. Aparecida - Padroeira do Brasil
02/11/2013	Finados
15/11/2013	Proclamação da República
25/12/2013	Natal

Scripts de Inclusão de Dados nas Tabelas mais Externas

USE Clinica;

GO

-- ESPECIALIDADE

INSERT INTO Especialidade (Descricao)

VALUES ('Clínico Geral'),

('Cardiologista'),

('Pediatria'),

('Ortopedista'),

('Neurologista');

SELECT * FROM Especialidade;

-- MEDICO

INSERT INTO Medico (CRM, Nome, Telefone)

VALUES ('100001', 'José Pedro Buarque' , '3012-4212'),

('100002', 'Maria Luiza Pontes' , '3123-5323'),

('100003', 'Silvana Maria Fraga' , '3234-6434'),

```
('100004','Marisilva Maria Santos','3345-7545'),
('100005','Maria José da Silva' , '3456-8656'),
('100006','Carlos André Nogueira' , '3567-9767'),
('100007','Ana Cristina Vargas' , '3678-0878'),
('100008','Manoel Carlos da Silva','3789-1989'),
('100009','João Batista Santos' , '3890-1090'),
('100010','Cristiana Silva Santos','3901-1101');
```

```
SELECT * FROM Medico;
```

```
-----
-- MEDICO_ESPECIALIDADE
-----
```

```
INSERT INTO Medico_Especialidade (CRM,Cd_Especialidade)
```

```
VALUES ('100001',1),
```

```
        ('100001',2),
```

```
        ('100002',3),
```

```
        ('100002',4),
```

```
        ('100003',2),
```

```
        ('100003',3),
```

```
        ('100003',5),
```

```
        ('100004',1),
```

```
        ('100004',2),
```

```
        ('100005',4),
```

```
        ('100005',5),
```

```
        ('100006',2),
```

```
        ('100006',3),
```

```
        ('100007',1),
```

```
        ('100008',3),
```

```
        ('100008',4),
```

```
        ('100009',5),
```

```
        ('100010',2),
```

```
        ('100010',4);
```

```
SELECT * FROM Medico_Especialidade;
```

-- DISPONIBILIDADE_MEDICO

INSERT INTO DisponibilidadeMedico
(CRM,DiaSemana,HoraInicial,HoraFinal,Situacao,PermiteEncaixe)

VALUES ('100001',2,'08:00','12:00',1,1),
('100001',4,'08:00','12:00',1,0),
('100001',6,'08:00','12:00',1,1),
('100002',2,'13:00','17:00',1,0),
('100002',4,'13:00','17:00',1,1),
('100002',6,'13:00','17:00',1,0),
('100003',3,'08:00','12:00',1,1),
('100003',5,'08:00','12:00',1,0),
('100003',7,'08:00','12:00',0,0),
('100004',3,'13:00','17:00',1,0),
('100004',5,'13:00','17:00',1,0),
('100004',7,'13:00','17:00',1,0),
('100005',2,'08:00','12:00',1,1),
('100005',4,'08:00','12:00',1,1),
('100005',6,'08:00','12:00',1,1),
('100006',2,'13:00','17:00',1,0),
('100006',4,'13:00','17:00',1,1),
('100006',6,'13:00','17:00',1,0),
('100007',3,'08:00','12:00',1,1),
('100007',5,'08:00','12:00',1,1),
('100008',3,'13:00','17:00',1,0),
('100008',5,'13:00','17:00',1,0),
('100009',4,'08:00','12:00',1,0),
('100009',6,'08:00','12:00',1,1),
('100010',5,'13:00','17:00',1,1),
('100010',7,'13:00','17:00',1,1);

SELECT * FROM DisponibilidadeMedico;

-- FERIADOS

INSERT INTO Feriado (Dt_Feriado,Descricao)

VALUES ('01/01/2013','Confraternização Universal'),
('12/02/2013','Carnaval'),
('29/03/2013','Paixão de Cristo'),
('21/04/2013','Tiradentes'),
('01/05/2013','Dia do Trabalho'),
('30/06/2013','Corpus Christi'),
('07/09/2013','Independência do Brasil'),
('12/10/2013','Nossa Sra. Aparecida - Padroeira do Brasil'),
('02/11/2013','Finados'),
('15/11/2013','Proclamação da República'),
('25/12/2013','Natal');

SELECT * FROM Feriado;

-- CLIENTES

INSERT INTO Cliente (Nome,Nascimento,Endereco,RG,NomeResponsavel,RGResponsavel) VALUES ('José Batista da Silva','20/02/2002','Rua das Flores','1234567','Maria José da Silva','7654321')

--SELECT FORMAT(Dt_Feriado,'d','en-gb') FROM Feriado

--SELECT CONVERT(CHAR(10),Dt_feriado,103), Descricao, DATEPART(DW,Dt_Feriado) FROM Feriado

--TRUNCATE TABLE Especialidade

--DELETE FROM Especialidade

Em relação aos modelos acima faça:

1. Crie todas as tabelas do modelo;
2. Insira todos os dados;
3. Crie um procedimento de nome **SP_GerarAgendaMedico** com o objetivo de gerar a agenda médica. Este procedimento deverá receber como parâmetro, Data da Agenda e CRM do Médico. Deverão ser tomados os seguintes cuidados:
 - 3.1. Se a data não é feriado;
 - 3.2. Se já não foi feita a geração da agenda para esta data e este médico;
 - 3.3. Se o médico atende neste dia da semana;
 - 3.4. Gerar os registros com diferença de 10 minutos entre a HoraInicial e HoraFinal conforme determinados na tabela de **Disponibilidade_Medico**.
4. Crie um procedimento de nome **SP_MarcarConsulta** com o objetivo de marcar consulta na agenda médica. Este procedimento deverá receber como parâmetro, **Cd_Cliente**, **CRM**, **Data** e **Hora** da marcação. Deverão ser tomados os seguintes cuidados:
 - 4.1. Verificar se o Cliente existe. Caso não exista, deverá ser apresentada a mensagem **"Cliente não cadastrado!"**;
 - 4.2. Verificar se o Médico existe. Caso não exista, deverá ser apresentada a mensagem **"Médico não cadastrado!"**;
 - 4.3. Verificar se existe a disponibilidade da data solicitada. Caso não exista, deverá ser apresentada a mensagem **"Agenda não disponível para esta data! "**
 - 4.4. Verificar se existe a disponibilidade de horário. Caso não exista, deverá ser apresentada a mensagem **"Horário indisponível! "**
 - 4.5. Apresentar a mensagem **"Consulta marcada! "**, caso a operação seja concluída.
 - 4.6. Verificar se a consulta já foi agendada para outro cliente, caso verdadeiro, emitir a mensagem **"Horário já reservado"**
 - 4.7. Verificar se existe a possibilidade de encaixe. Caso exista, deverá ainda ser verificado se já não foi feito encaixe nesta hora. Caso não exista a possibilidade de encaixe, deverá ser apresentada a mensagem **"Encaixe negado! "**, caso contrário, deverá ser feito o encaixe, registrado na agenda a opção **Encaixe=1** e emitir a mensagem **"Encaixe marcado"**.
5. Faça os testes necessários ao perfeito funcionamento dos procedimentos.

Exemplos de Testes:

Gerando a agenda

```
EXEC SP_GerarAgendaMedico '01/10/2013', '100003'; EXEC  
SP_GerarAgendaMedico '03/10/2013', '100003';
```

Marcando Consultas

Verifica se o cliente existe (4.1)

EXEC SP_MarcarConsulta 0,'100011','02/10/2013','08:15';	
Resultado:	Mostra Mensagem “ Cliente não cadastrado! ”
Motivo:	O Cliente não possui cadastro.

Verifica se o médico existe (4.2)

EXEC SP_MarcarConsulta 1,'100011','02/10/2013','08:15';	
Resultado:	Mostra Mensagem “ Medico não cadastrado! ”
Motivo:	O Médico não possui cadastro.

Verifica a disponibilidade de data (4.3)

EXEC SP_MarcarConsulta 1,'100003','02/10/2013','08:15'	
Resultado:	Mostra Mensagem “ Agenda não disponível para esta data! ”
Motivo:	O Médico não atende neste dia da semana.

Verifica a disponibilidade de horário (4.4)

EXEC SP_MarcarConsulta 1,'100003','03/10/2013','08:15'	
Resultado:	Mostra Mensagem: “ Horário Indisponível! ”
Motivo:	A agenda deste médico não permite encaixe neste dia da semana.

Marca a consulta (4.5)

EXEC SP_MarcarConsulta 1,'100003','01/10/2013','08:10'	
Resultado:	Mostra Mensagem: “ Consulta Marcada! ”
Motivo:	A agenda deste médico disponível para esta data.

Verifica se horário está disponível (4.6)

EXEC SP_MarcarConsulta 1,'100003','01/10/2013','08:10'	
Resultado:	Mostra Mensagem Horário já reservado!
Motivo:	Data já utilizada para este Médico.

Marca consulta por encaixe (4.7)

EXEC SP_MarcarConsulta 1,'100003','01/10/2013','08:15'	
Resultado:	“ Encaixe marcado! ”
Motivo:	Este Médico permite encaixe neste dia da semana.

Verifica se é possível novo encaixe (4.7)

EXEC SP_MarcarConsulta 1,'100003','01/10/2013','08:18'	
Resultado:	Mostra Mensagem “ Encaixe negado! ”

Motivo:	Já houve outro encaixe nesta hora (>=8h e <9h).
---------	---

6. Desenvolva uma função chamada **F_DiaSem** capaz de mostrar o dia da semana conforme mostrado abaixo:

```
SELECT CRM, HoraInicial, HoraFinal, Situacao, PermiteEncaixe, dbo.F_DiaSem(DiaSemana) AS 'Dia' FROM
DiagnosticoMedico
WHERE CRM='100003'
```

	CRM	HoraInicial	HoraFinal	Situacao	PermiteEncaixe	Dia
1	100003	08:00:00	12:00:00	1	1	TER
2	100003	08:00:00	12:00:00	1	0	QUI
3	100003	08:00:00	12:00:00	0	0	SAB

7. Desenvolva um trigger chamado **TR_Marcacao** ligado a tabela **AgendaMedico**. O trigger deverá ser acionado quando da marcação de uma consulta. Ao ser chamado, deverá verificar se o campo situação era igual a um, antes da marcação, e caso positivo, deverá salvar na tabela **AgendaControle**, os dados de antes e depois da alteração como as respectivas data e hora em que ela ocorreu.

AgendaControle

CAMPO	TIPO	CHAVE	NULO
Nr_Movimento	INTEGER IDENTITY		
Cd_AgendaMedico	INTEGER		
CRM	VARCHAR(20)		
Cd_Cliente	INTEGER		
Data	DATE		
Hora	TIME(0)		
Situacao	BIT		
Encaixe	BIT		
DataMovimentacao	DATE		
HoraMovimentacao	TIME(0)		

Testando Movimentação

```
UPDATE AgendaMedico  
SET     Cd_cliente=null, Situacao=0  
WHERE   Cd_AgendaMedico=2
```