

1)

**R:** *Eu diria que stored procedure são funções que podem ser programadas para executar de terminada tarefa a associada e a campos, tabelas, valores de campos entre outros. E sua essência esta no reaproveitamento de código, uma vez criada pode ser chamada quantas vezes forem necessárias.*

2)

```
go
create procedure inserir_departamento (@ID int, @nome varchar(100))
as
begin
    insert into departamentos
        (id, nome) values
        (@ID, @nome)
end
```

3)

```
create procedure atualizar_o_status
as
begin
    declare @cliente_id DECIMAL
    declare @cliente_consumo DECIMAL

    DECLARE cursor_cliente_id CURSOR FOR
    select id, consumo from clientes

    open cursor_cliente_id
    FETCH cursor_cliente_id INTO @cliente_id, @cliente_consumo

    WHILE @@FETCH_STATUS = 0
    BEGIN
        begin
            if @cliente_consumo < 100
                update clientes set status = 'C' WHERE id = @cliente_id
            end
        begin
            if @cliente_consumo > 100 and @cliente_consumo < 150
                update clientes set status = 'B' WHERE id = @cliente_id
            end
        begin
            if @cliente_consumo > 150
                update clientes set status = 'A' WHERE id = @cliente_id
            end

            FETCH cursor_cliente_id INTO @cliente_id, @cliente_consumo
        end

        CLOSE cursor_cliente_id
        DEALLOCATE cursor_cliente_id
    end
```

4)

```
go
Create procedure nome_aluno (@MATRICULA DECIMAL, @nome_sobrenome VARCHAR(50)
OUT)
as
begin
    declare @ALUNO_NOME VARCHAR(20)
    declare @ALUNO_SOBRENOME VARCHAR(30)

    select @ALUNO_NOME = NOME FROM ALUNO
    select @ALUNO_SOBRENOME = SOBRENOME FROM ALUNO

    SET @nome_sobrenome = CONCAT(@ALUNO_NOME,@ALUNO_SOBRENOME)

end
```

5)

```
create table Cadastro(
    Matricula int not null primary key,
    Nome varchar(30) not null,
    Email varchar(40) null,
    DtaNascto date not null,
    CPF decimal(11) not null)

create table Log(
    id int not null identity,
    DtaHora datetime not null,
    TipoExecucao char(01) not null, --I / A / D
    NomeUsuario varchar(30) not null,
    HostName varchar(30) not null,
    ValoresInseridos varchar(200) null,
    ValoresExcluidos varchar(200) null)

select getdate()
select SYSTEM_USER
select host_name()

select * from Cadastro

insert into Cadastro (Matricula, Nome, Email, DtaNascto, CPF)
values (4321, 'Wesley', 'wesleyafdp@gmail.com', '1991-03-05', '256969')

select cast(Matricula as varchar)+';' from Cadastro

GO
CREATE TRIGGER TGR_Cadastro_INSERT
ON Cadastro
FOR INSERT
AS
BEGIN
    DECLARE
        @Matricula int,
        @Nome varchar(30),
        @Email varchar(40),
        @DtaNascto date,
        @CPF decimal(11),
```

```

-----
@DtaHora datetime = cast(getdate() as varchar),
@TipoExecucao char(1) = 'I',
@NomeUsuario varchar(30) = SYSTEM_USER,
@HostName varchar(30) = host_name()

SELECT
    @Matricula = Matricula,
    @Nome = Nome,
    @Email = Email,
    @DtaNascto = DtaNascto,
    @CPF = CPF
FROM inserted

DECLARE
    @ValoresInseridos varchar(900) = concat(@Matricula,';',@Nome,';',
@Email,';',@DtaNascto ,';',@CPF)

insert into Log (
    DtaHora,
    TipoExecucao,
    NomeUsuario,
    HostName,
    ValoresInseridos
)
Values(
    @DtaHora,
    @TipoExecucao,
    @NomeUsuario,
    @HostName,
    @ValoresInseridos
)

END
go

```

```

-----
CREATE TRIGGER TGR_Cadastro_DELETE
ON Cadastro
FOR DELETE
AS
BEGIN
    DECLARE
        @Matricula int,
        @Nome varchar(30),
        @Email varchar(40),
        @DtaNascto date,
        @CPF decimal(11),
        @DtaHora datetime = cast(getdate() as varchar),
        @TipoExecucao char(1) = 'D',
        @NomeUsuario varchar(30) = SYSTEM_USER,
        @HostName varchar(30) = host_name()

    SELECT
        @Matricula = Matricula,
        @Nome = Nome,

```

```

        @Email = Email,
        @DtaNascto = DtaNascto,
        @CPF = CPF
FROM deleted

DECLARE
    @ValoresExcluidos varchar(900) = concat(@Matricula,';',@Nome,';',
@Email,';',@DtaNascto ,';',@CPF)

```

```

insert into Log (
    DtaHora,
    TipoExecucao,
    NomeUsuario,
    HostName,
    ValoresExcluidos
)
Values(
    @DtaHora,
    @TipoExecucao,
    @NomeUsuario,
    @HostName,
    @ValoresExcluidos
)

END
go

```

```

-----
CREATE TRIGGER TGR_Cadastro_UPDATE
ON Cadastro
FOR UPDATE
AS
BEGIN
    DECLARE

```

```

        @Matricula int,
        @Nome varchar(30),
        @Email varchar(40),
        @DtaNascto date,
        @CPF decimal(11),

        -----
        @DtaHora datetime = cast(getdate() as varchar),
        @TipoExecucao char(1) = 'A',
        @NomeUsuario varchar(30) = SYSTEM_USER,
        @HostName varchar(30) = host_name()

```

```

SELECT
    @Matricula = Matricula,
    @Nome = Nome,
    @Email = Email,
    @DtaNascto = DtaNascto,
    @CPF = CPF
FROM deleted

```

```

DECLARE
    @ValoresExcluidos varchar(900) = concat(@Matricula,';',@Nome,';',
@Email,';',@DtaNascto ,';',@CPF)

```

```

SELECT
    @Matricula = Matricula,
    @Nome = Nome,
    @Email = Email,
    @DtaNascto = DtaNascto,
    @CPF = CPF
FROM inserted

DECLARE
    @ValoresInseridos varchar(900) = concat(@Matricula,';',@Nome,';',
@Email,';',@DtaNascto ,';',@CPF)

insert into Log (
    DtaHora,
    TipoExecucao,
    NomeUsuario,
    HostName,
    ValoresInseridos,
    ValoresExcluidos
)
Values(
    @DtaHora,
    @TipoExecucao,
    @NomeUsuario,
    @HostName,
    @ValoresInseridos,
    @ValoresExcluidos
)

END
go

```