

```

1  -----
2  ----- CREATE SOME Stored Procedure -----
3  --                                     and use them...
4  -----
5
6
7  -----
8  -- USE MEI_TRAB: Changes the database context to the MEI_TRAB database.
9  --
10 USE MEI_TRAB
11 --
12 -----
13
14 IF EXISTS (SELECT name FROM sysobjects
15            WHERE name = 'INSERIR_ENCOMENDA' AND type = 'P')
16     DROP PROCEDURE INSERIR_ENCOMENDA
17 GO
18
19 -- Insere uma nova encomenda
20
21 CREATE PROCEDURE INSERIR_ENCOMENDA
22 AS
23     DECLARE @EncId int
24
25     DECLARE @MAX_TRY int
26     DECLARE @NTry int
27
28     SET @MAX_TRY = 20;    -- Max number of times to try
29
30     SET @NTry = 1;
31
32     -- try to insert the new Encomenda
33     WHILE (@NTry <= @MAX_TRY)
34     BEGIN
35         Set @NTry = @NTry +1;
36
37         -- get next EncId
38         Select @EncId = Max(EncId) From Encomenda;
39
40         IF (@EncId IS NULL)
41             SET @EncId = 1;
42         ELSE
43             SET @EncId = @EncId +1;
44
45         -- Inserir encomenda
46         INSERT INTO Encomenda Values (@EncId, 1000, 'Fernando Pessoa', 'Lisboa');
47
48         IF (@@ERROR = 0)    -- Success!
49         BEGIN
50             -- Inserir linhas da encomenda
51             INSERT INTO EncLinha Values (@EncId, 111, 'Mensagem', 2500, 2);
52             INSERT INTO EncLinha Values (@EncId, 131, 'Livro do Desassossego', 3000, 1);
53
54             BREAK  -- Sucesso, hora de sair!
55         END
56     END  -- Fim de ciclo
57
58     IF ( @NTry > @MAX_TRY)
59     BEGIN
60         RAISERROR('Can't insert the record',16,1 )
61         ROLLBACK TRANSACTION
62     END
63 GO
64
65
66 IF EXISTS (SELECT name FROM sysobjects
67            WHERE name = 'APAGAR_ENCOMENDA' AND type = 'P')
68     DROP PROCEDURE APAGAR_ENCOMENDA
69 GO
70
71 -- Apaga uma encomenda

```

```

72
73 CREATE PROCEDURE APAGAR_ENCOMENDA
74 AS
75     DECLARE @EncId int
76
77     DECLARE @N_Linhas int
78
79     -- Obter a quantidade de encomendas
80     Select @N_Linhas = Count (*) From Encomenda
81
82     DECLARE @Random INT, @Upper INT, @Lower INT
83
84     -- Limites para a geração de números aleatórios
85     Set @Lower = 1; -- Menor valor
86     Set @Upper = @N_Linhas; -- Maior valor
87
88     -- Escolher aleatoriamente a linha a apagar
89     Set @Random = ROUND(((@Upper - @Lower - 1) * RAND() + @Lower), 0);
90
91     -- Obter o ID da encomenda na linha "Random"
92     Select @EncId = EncId
93     from ( SELECT Row_Number() Over (Order By EncId) as N_Linha, [EncID]
94           FROM Encomenda) AS X
95     Where N_Linha = @Random
96
97     Delete From EncLinha Where EncId = @EncId
98     Delete From Encomenda Where EncId = @EncId
99
100 GO
101
102
103 -- Actualiza uma encomenda
104 IF EXISTS (SELECT name FROM sysobjects
105           WHERE name = 'ACTUALIZAR_ENCOMENDA' AND type = 'P')
106     DROP PROCEDURE ACTUALIZAR_ENCOMENDA
107 GO
108
109 CREATE PROCEDURE ACTUALIZAR_ENCOMENDA
110 AS
111     DECLARE @EncId int
112
113     DECLARE @N_Linhas int
114
115     -- Obter a quantidade de encomendas
116     Select @N_Linhas = Count (*) From Encomenda
117
118     DECLARE @Random INT, @Upper INT, @Lower INT
119
120     -- Limites para a geração de números aleatórios
121     Set @Lower = 1; -- Menor valor
122     Set @Upper = @N_Linhas; -- Maior valor
123
124     -- Escolher aleatoriamente a linha a encomendar
125     Set @Random = ROUND(((@Upper - @Lower - 1) * RAND() + @Lower), 0);
126
127     -- Obter o ID da encomenda na linha "Random"
128     Select @EncId = EncId
129     from ( SELECT Row_Number() Over (Order By EncId) as N_Linha, [EncID]
130           FROM Encomenda) AS X
131     Where N_Linha = @Random
132
133     -- Alterar o campo Morada para o instante corrente
134     Update Encomenda
135         Set Morada = CONVERT(varchar, GETDATE(), 121)
136     Where EncId = @EncId
137
138 GO
139

```