

Nome do Aluno:

RA:

Turma:

Série:

Data: __/__/____

Professor(a) Responsável: Gustavo Bianchi Maia

Nota:

(

)

Visto do Prof.

Seu objetivo é automatizar os processos de armazenamento de logins, senhas e dicas de senha e o processo de autenticação de um sistema, de forma que a utilização das informações deve se dar da seguinte maneira:

- Uma tabela TBL_CTRL_ACESSO, com os campos citados acima, com
 - Login, não precisa nem deve ser criptografado na base
 - Senha, deve ser armazenado com criptografia 1-WAY
 - Dica_senha, deve ser armazenado com criptografia 2-WAY

Para concluir esta atividade você precisará criar:

- Uma função, que receba um texto e devolva-o criptografado (fn_encrypt)
- Outra Função, que receba um valor criptografado e o decriptografe (fn_decrypt)
- Uma função, que receba um texto e o criptografe (fn_hash) □ use "SALT"

De forma que tais funções possam ser utilizadas nos exemplos criados para insirir valores na tabela obedecendo às recomendações.

Finalmente:

- Crie uma procedure que: dado um login e senha, devolva "1" se autenticado e "0" caso contrário.
- Crie uma procedure que: dado um login, devolva a dica da senha, decriptografada.

Preencha esta ficha de avaliação, copie e cole todos os códigos utilizados e entregue para o professor.

Abaixo algumas orientações passadas em sala de aula:

Sejam as seguintes funções (Que serão criadas na AC2)

--Função de criptografia

```
SELECT dbo.Fn_encrypt('oi')
```

Recebe uma string e devolve o respectivo valor criptografado em VARBINARY utilizando um algoritmo de 2 vias assimétrico de criptografia.

--Função de decriptografia

```
SELECT dbo.Fn_decrypt(dbo.Fn_encrypt('oi'))
```

Recebe um valor já criptografado e devolve o respectivo valor descriptografado já convertido em VARCHAR

--Função de criptografia de HASH

```
SELECT dbo.Fn_hash('oi')
```

Recebe uma string e devolve o respectivo valor criptografado em VARBINARY utilizando um algoritmo de HASH com a utilização de um SALT (por segurança).

--Criando a tabela TBL_CTRL_ACESSO

```
CREATE TABLE tbl_ctrl_acesso
( [login]          VARCHAR(60) NOT NULL,
  [senha]          VARBINARY(max) NOT NULL,
  [dica_senha]     VARBINARY(max) NULL,
  CONSTRAINT pk_ctrl_acesso PRIMARY KEY ( [login] )
)
```

--Inserindo valores nas tabelas para testes:

```
INSERT INTO tbl_ctrl_acesso( [login], [senha], [dica_senha] )
VALUES      ( 'José', dbo.Fn_hash('senha'),dbo.Fn_encrypt('aquela lá') )
go
```

--Testando valores brutos inseridos na tabela

```
SELECT * FROM   tbl_ctrl_acesso
go
```

--Testando valores decriptografados lidos da tabela

```
SELECT [login], [senha],
       CONVERT(VARCHAR, dbo.Fn_decrypt([dica_senha])) AS [dica_senha]
FROM   tbl_ctrl_acesso
go
```

Seja a seguinte procedure (que também será criada na AC2)

```
EXEC Pr_login @login, @senha, @result output
```

Que, recebe um login e senha (ambos varchar) e devolve 1 se ele foi autenticado, ou seja, se aquele usuário foi cadastrado com aquela senha, e 0 caso contrário.

Exemplo de utilização:

--testando procedure de Login

```
DECLARE @result BIT
```

--autenticado

```
EXEC Pr_login 'josé', 'senha', @result output
```

```
SELECT CASE WHEN @result = 1 THEN 'Autenticado' ELSE 'Não autenticado' END
```

--não autenticado

```
EXEC Pr_login 'josé', 'senha errada', @result output
```

```
SELECT CASE WHEN @result = 1 THEN 'Autenticado' ELSE 'Não autenticado' END
```

```
go
```

Seja a seguinte procedure (que também será criada na AC2)

```
EXEC Pr_esqueci_senha @login, @result output
```

Que, recebe um login (varchar) e devolve a dica da senha decriptografada, cadastrada para aquele login.

Exemplo de utilização:

--Testando a procedure esqueci senha

```
DECLARE @result VARCHAR(60)
```

```
EXEC Pr_esqueci_senha 'josé', @result output
```

```
SELECT 'Sua dica da senha é: '' + @result + ''
```

```
go
```

Observações:

- Avaliação em duplas.
- Organização, clareza e legibilidade na resolução da prova serão consideradas na correção.
- A interpretação da atividade faz parte do processo de avaliação
- Qualquer tentativa de fraude, cola será punido com nota zero.
- O total desta avaliação corresponde à 1 de 10 avaliações continuadas.

Boa Avaliação!