O **md5** trata-se de um algoritmo de criptografia unidirecional, assim não pode ser descriptografado, Dessa forma, é excelente para se gravar senhas.

A biblioteca **JAVA** que possui a criptografia **md5** é a biblioteca **java.security.MessageDigest** o MessageDigest é uma classe usada para criptografar textos com a criptografia md5.

Para criarmos uma instância de um MessageDigest utilizamos o método **getInstance()**.

MessageDigest md = MessageDigest.getInstance(“MD5”);

Após criarmos o objeto para criptografar e especificar o algoritmo **MD5**, precisamos gerar a chave criptografada com o método **digest()**, ele possui três tipos de assinaturas:

**byte[] digest();**

**byte[] digest(byte[] input);**

**int digest(byte[] buf, int offset, int len) trhows DigestException;**

Classe:

import java.security.MessageDigest;

import java.security.NoSuchAlgorithmException;

public class Criptografia {

private static MessageDigest md = null;

static {

try {

//Cria o objeto com a instancia de criptografia no padrão MD5

md = MessageDigest.getInstance("MD5");

} catch (NoSuchAlgorithmException ex) {

ex.printStackTrace();

}

}

//Método para criar a chave criptografada

private static char[] hexCodes(byte[] text) {

char[] hexOutput = new char[text.length \* 2];

String hexString;

for (int i = 0; i < text.length; i++) {

hexString = "00" + Integer.toHexString(text[i]);

hexString.toUpperCase().getChars(hexString.length() - 2, hexString.length(), hexOutput, i \* 2);

}

return hexOutput;

}

//Método para criptografar o texto

public static String criptografar(String pwd) {

if (md != null) {

return new String(hexCodes(md.digest(pwd.getBytes())));

}

return null;

}

}