Practice 03:

Given two strings s and t, *determine if they are isomorphic*.

Two strings s and t are isomorphic if the characters in s can be replaced to get t.

All occurrences of a character must be replaced with another character while preserving the order of characters. No two characters may map to the same character, but a character may map to itself.

**Example 1:**

**Input:** s = "egg", t = "add"

**Output:** true

**Example 2:**

**Input:** s = "foo", t = "bar"

**Output:** false

**Example 3:**

**Input:** s = "paper", t = "title"

**Output:** true

**Constraints:**

* 1 <= s.length <= 5 \* 104
* t.length == s.length
* s and t consist of any valid ascii character.

Code:

class Solution {

public boolean isIsomorphic(String s, String t) {

int[] mappingDictStoT = new int[256];

Arrays.fill(mappingDictStoT, -1);

int[] mappingDictTtoS = new int[256];

Arrays.fill(mappingDictTtoS, -1);

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

char c1 = s.charAt(i);

char c2 = t.charAt(i);

if (mappingDictStoT[c1] == -1 && mappingDictTtoS[c2] == -1) {

mappingDictStoT[c1] = c2;

mappingDictTtoS[c2] = c1;

}

else if (!(mappingDictStoT[c1] == c2 && mappingDictTtoS[c2] == c1)) {

return false;

}

}

return true;

}

}