Convert String to Lower Case.

This question is not frequently asked in interviews but it is good learning exercise to implement the solution. Also, I would like to remind that we are not supposed to use toLowerCase() method of String API.

We can leverage the concept of ASCII.

Example:

A-65, B-66, .., Z=90

a-97, b-98, .. ,z=122

Link for ASCII table, click [here](http://www.asciitable.com/) .

You can generate ASCII table using simple program.

**for** (**int** i = 1; i <= 256; i++) {

System.***out***.println(i+" "+(**char**)i);

}

Now we have ASCII for all characters with us. Now a simple idea to convert a string to lower case is to replace the old ASCII with new ASCII value.

For example:

A is 65 and a is 97.

Now we have input as “A”.

Subtract 65 and add 97 to the character we are working on so now the value is 97. And 97 is ASCII for “a”.

Let us try another example.

“D” is 68 and d is 100.

68-65 = 3

Add 97 to the difference i.e 97+3=100

Subtract 65 and add 97 to the character we are working on so now the value is 100. And 100 is ASCII for “d”.

Well we now have our solution. Also one more thing we do not touch any other characters whose ASCII is < 65 and > 90. We leave them alone.

Let us design the test cases for this problem.

|  |  |
| --- | --- |
| Input | Output |
| null | null |
| “a” | “a” |
| “A” | “a” |
| “laptop” | “laptop” |
| “LAPTOP” | “laptop” |
| “password@111” | “password@111” |
| “PASSWORD@111” | “password@111” |
| “password@111” | “password@111” |

Test Cases are as follows:

@Test

**public** **void** testStringToLowerForNull() {

Assert.*assertEquals*(**null**, StringToLowerCase.*stringToLowerCase*(**null**));

}

@Test

**public** **void** testStringToLowerForSingleLengthLowerCase() {

Assert.*assertEquals*("a", StringToLowerCase.*stringToLowerCase*("a"));

}

@Test

**public** **void** testStringToLowerSingleLengthUpperCase() {

Assert.*assertEquals*("a", StringToLowerCase.*stringToLowerCase*("A"));

}

@Test

**public** **void** testStringToLowerForLowerCase() {

Assert.*assertEquals*("laptop", StringToLowerCase.*stringToLowerCase*("laptop"));

}

@Test

**public** **void** testStringToLowerForUpperCase() {

Assert.*assertEquals*("laptop", StringToLowerCase.*stringToLowerCase*("LAPTOP"));

}

@Test

**public** **void** testStringToLowerAdditionalCharacterLowerCase() {

Assert.*assertEquals*("password@111", StringToLowerCase.*stringToLowerCase*("password@111"));

}

@Test

**public** **void** testStringToLowerAdditionalCharacterUpperCase() {

Assert.*assertEquals*("password@111", StringToLowerCase.*stringToLowerCase*("PASSWORD@111"));

}

@Test

**public** **void** testStringToLowerAdditionalCharacter() {

Assert.*assertEquals*("password@111", StringToLowerCase.*stringToLowerCase*("passWoRD@111"));

}

Now let us write code:

Check for null.

/\*\*

\* If string is null then return null

\* \*/

**if** (str == **null**) {

**return** **null**;

}

Check for 0 length.

/\*\*

\* If the length of string is 0 then return it.

\* \*/

**else** **if** (str.length() == 0) {

**return** str;

}

Search for UPPER case letters and convert it to lower case.

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

/\*\*

\* We make this check so as to convert UPPER case letters to lower case.

\* range for UPPER case is from 65 to 90.

\* \*/

**if** (str.charAt(i) >= 65 && str.charAt(i) <= 90) {

newStr += (**char**) (str.charAt(i) - 65 + 97);

}

/\*\*

\* If it is not UPPER case then just copy that character into new string.

\* \*/

**else** {

newStr += str.charAt(i);

}

}

Code for entire method is as follows.

**public** **static** String stringToLowerCase(String str) {

/\*\*

\* If string is null then return null

\* \*/

**if** (str == **null**) {

**return** **null**;

}

/\*\*

\* If the length of string is 0 then return it.

\* \*/

**else** **if** (str.length() == 0) {

**return** str;

}

// declare new string to store data into it.

String newStr = "";

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

/\*\*

\* We make this check so as to convert UPPER case letters to lower case.

\* range for UPPER case is from 65 to 90.

\* \*/

**if** (str.charAt(i) >= 65 && str.charAt(i) <= 90) {

newStr += (**char**) (str.charAt(i) - 65 + 97);

}

/\*\*

\* If it is not UPPER case then just copy that character into new string.

\* \*/

**else** {

newStr += str.charAt(i);

}

}

**return** newStr;

}