1. **Reverse Split**

Write a program to read a string and a character, and reverse the string and convert it in a format such that each character is separated by the given character. Print the final string.

Include a class **UserMainCode** with a static method **reshape**which accepts a string and a character. The return type (String) should return the final string.

Create a Class Main which would be used to accept a string and a character, and call the static method present in UserMainCode.

**Input and Output Format:**

Input consists of a string and a character.

Output consists of a string (the final string).

Refer sample output for formatting specifications.

**Sample Input:**

Rabbit

-

**Sample Output:**

t-i-b-b-a-R

1. **String Occurances - II**

Obtain two strings S1,S2 from user as input. Your program should count the number of times S2 appears in S1.  
  
Return the count as output. Note - Consider case.  
  
Include a class UserMainCode with a static method **getSubstring** which accepts two string variables. The return type is the count.  
  
Create a Class Main which would be used to accept two Input strings and call the static method present in UserMainCode.  
  
**Input and Output Format:**  
  
Input consists of two strings with maximum size of 100 characters.  
  
Output consists of an integer.  
  
Refer sample output for formatting specifications.  
  
**Sample Input 1:**  
catcowcat  
cat  
  
**Sample Output 1:**  
2  
  
  
**Sample Input 2:**  
catcowcat  
CAT  
  
**Sample Output 2:**  
0

1. **Repeat Front**

Given a string (s) and non negative integer (n) apply the following rules.

1. Display the first three characters as front.
2. If the length of the string is less than 3, then consider the entire string as front and repeat it n times.

Include a class UserMainCode with a static method **repeatFirstThreeCharacters** which accepts the string and integer. The return type is the string formed based on rules.  
  
Create a Class Main which would be used to accept the string and integer and call the static method present in UserMainCode.  
  
**Input and Output Format:**  
  
Input consists of a string and integer.  
  
Output consists of a string .  
  
Refer sample output for formatting specifications.  
  
**Sample Input 1:**  
Coward  
2  
  
**Sample Output 1:**  
CowCow  
  
  
**Sample Input 2:**  
So  
3  
  
**Sample Output 2:**  
SoSoSo

1. **Playing with String - I**

Given a string array and non negative integer (n) apply the following rules.  
  
1. Pick nth character from each String element in the String array and form a new String.  
2. If nth character not available in a particular String in the array consider $ as the character.  
3. Return the newly formed string.  
  
Include a class UserMainCode with a static method **formString** which accepts the string and integer. The return type is the string formed based on rules.  
  
Create a Class Main which would be used to accept the string and integer and call the static method present in UserMainCode.  
  
**Input and Output Format:**  
  
Input consists of a an integer which denotes the size of the array followed by the array of strings and an integer (n).  
Output consists of a string .  
Refer sample output for formatting specifications.  
  
**Sample Input 1:**  
4  
ABC  
XYZ  
EFG  
MN  
3

**Sample Output 1:**  
CZG$