1. Read from a terminal using scanf function and print using printf function.

```c
#include<stdio.h>
int main(){
        char name[20];
   printf("Enter name: ");
   scanf("%s", name);
   printf("Your name is %s.", name);
   return 0;
}
```

OUTPUT:

Enter name: Sreyash Mishra

Your name is Sreyash.

2. Read a lines of text from a terminal using fgets function and print using puts function.

```c
#include<stdio.h>
int main(){
        char name[20];
   printf("Enter name: ");
   fgets(name,sizeof(name),stdin);
   printf("name: ");
   puts(name);
   return 0;
}
```

OUTPUT:

Enter name: Sreyash Mishra

name: Sreyash Mishra

3. Convert

a. Upper case to Lower case

b. Lower case to Upper case

c. Toggle case

d. Sentence case


```c
//LOWER CASE
#include <stdio.h>
#include <string.h>
int main(){
  char s[100];
  int i;
  printf("Enter a string : ");
  gets(s);
  for (i = 0; s[i]!='\0'; i++) {
    if(s[i] >= 'A' && s[i] <= 'Z') {
      s[i] = s[i] + 32;
    }
  }
  printf("\nString in Lower Case = %s", s);
  return 0;
```

}

Enter a string : Sreyash

String in Lower Case = sreyash


```c
//UPPER CASE
#include <stdio.h>
#include <string.h>
int main() {
  char s[100];
  int i;
  printf("Enter a string : ");
  gets(s);
  for (i = 0; s[i]!='\0'; i++) {
    if(s[i] >= 'a' && s[i] <= 'z') {
      s[i] = s[i] - 32;
    }
  }
  printf("\nString in Upper Case = %s", s);
  return 0;
}
```


OUTPUT:

Enter a string : sreyash mishra

String in Upper Case = SREYASH MISHRA

//TOGGLE CASE

```c
#include <stdio.h>
#include <string.h>
int main(){
        char Str[100];
        int i;
        printf("Enter any string: ");
        gets(Str);
        for (i = 0; Str[i]!='\0'; i++){
                if(Str[i] >= 'a' && Str[i] <= 'z'){
                        Str[i] = Str[i] - 32;
                }
                else if(Str[i] >= 'A' && Str[i] <= 'Z'){
                Str[i] = Str[i] + 32;
        }
        }
        printf("\n The Given String after toggle case = %s", Str);
return 0;
}
```

OUTPUT:

Enter any string: HeLlO

 The Given String after toggle case = hElLo

```c
//SENTENCE CASE

#include <stdio.h>

#include <ctype.h>

int main(){

        char str[100];

        printf("Enter a string : ");

   gets(str);

   str[0] = toupper(str[0]);

   printf("The string is: %s.",str);

   return 0;

}
```

OUTPUT:

Enter a string : hello programmers

The string is: Hello programmers.

4. Perform String Concatenation (With and Without String Handling Functions).

//CONCATE WITHOUT FUNC

```c
#include <stdio.h>

int main() {

 char s1[100] = "Hello ", s2[] = "Happy Learning";

 int length, j;

 length = 0;

 while (s1[length] != '\0') {
```

```c
        ++length;

    }

    for (j = 0; s2[j] != '\0'; ++j, ++length) {

      s1[length] = s2[j];

    }

    s1[length] = '\0';

    printf("After concatenation: ");

    puts(s1);

return 0;

}
```

OUTPUT:

After concatenation: Hello Happy Learning

```c
//WITH FUNC
#include <stdio.h>
#include <string.h>
int main(){
    char str[100], str2[100];
    printf("Enter the first string\n");
    gets(str);
    printf("Enter the second string\n");
    gets(str2);
    strcat(str,str2);
    printf("String obtained on concatenation is %s\n",str);
```

```c
    return 0;

}
```

OUTPUT:

Enter the first string

HELLO

Enter the second string

WORLD

String obtained on concatenation is HELLOWORLD

5. Perform String Reversal (With and Without String Handling Functions).

//WITHOUT FUNC

```c
#include<stdio.h>

#include<string.h>

int main() {

  char str[100], temp;

  int i, j = 0;

 printf("Enter the string: ");

  gets(str);

  i = 0;

  j = strlen(str) - 1;

  while (i < j) {

    temp = str[i];

    str[i] = str[j];

    str[j] = temp;
```

```c
    i++;

    j--;

  }

  printf("\nReverse string is :%s", str);

return 0;

}
```

OUTPUT:

Enter the string: HELLO HI

Reverse string is :IH OLLEH


```c
//WITH FUNC

#include <stdio.h>

#include <string.h>

int main()

{

  char s[100];

  printf("Enter a string to reverse ");

  gets(s);

  strrev(s);

  printf("Reverse of the string: %s\n", s);

  return 0;

}
```

OUTPUT:

Enter the string: HI BYE

Reverse string is :EYB IH

6. Perform Substring Extraction (With and Without String Handling Functions).

//WITHOUT FUNC

```c
#include <stdio.h>

int main(){
  char str[100], sstr[100];
  int pos, l, c = 0;
    printf("Input the string : ");
    fgets(str, sizeof str, stdin);
  printf("Input the position to start extraction :");
  scanf("%d", &pos);
  printf("Input the length of substring :");
  scanf("%d", &l);
  while (c < l)
  {
    sstr[c] = str[pos+c-1];
    c++;
  }
  sstr[c] = '\0';
  printf(sstr);
}
```

OUTPUT:

Input the string : HELLOWORLD

Input the position to start extraction :5

Input the length of substring :6

OWORLD

7. Copy one string into another and count the no of elements copied. (With and Without String Handling Functions).

```c
#include<stdio.h>

//#define N 10

int main()

{

char str1[80], str2[80];

int i;

printf("Input a string: ");

scanf("%s", str2);

for(i=0; str2[i]!='\0'; i++)

str1[i]=str2[i];

str1[i]='\0';

printf("\n");

printf("Original string: %s", str1);

printf("\nNumber of characters = %d\n", i);

return 0;

}
```

OUTPUT:

Input a string: SREYASH

Original string: SREYASH

Number of characters = 7

```
//WITH FUNC

#include<stdio.h>

#include<string.h>   // for using strcpy() function

int main(){

    char str1[100];

    char str2[100];

    int i;

    printf("Enter the string: ");

    gets(str2);

    strcpy(str1,str2);

    printf("\nThe copied string is: %s", str1);

    for(i=0; str2[i]!='\0'; i++)

        str1[i]=str2[i];

        str1[i]='\0';

    printf("\nNumber of characters = %d\n", i);

    return 0;

}
```

OUTPUT:

Enter the string: Sreyash

The copied string is: Sreyash

Number of characters = 7

8. Read a string and prints if it is a palindrome or not.

```
#include <stdio.h>

#include <string.h>
```

```c
int main(){

char a[100], b[100];

printf("Enter the string :" );

gets(a);

strcpy(b, a); /* Copying input string */

strrev(b); /* Reversing the string */

if (strcmp(a, b) == 0) /* Comparing input string with the reverse string */

printf("The string is a palindrome\n");

else

printf("The string is not t a palindrome\n");

return 0;

}
```

OUTPUT:

Enter the string : my

string is not palindrome

Enter the string : wow

string is palindrome

9. Read a line of text and count all occurrences of particular word.

```c
#include<stdio.h>

#include <string.h>

int main(){

   char s[1000],w[1000];
```

```c
int n,a[1000],i,j,k=0,l,found=0,t=0;

printf("Enter  the string: ");

gets(s);

printf("Enter word to be searched: ");

gets(w);

for(i=0;s[i];i++)

{

        if(s[i]==' ')

        {

                a[k++]=i;

        }

}

a[k++]=i;

j=0;

for(i=0;i<k;i++)

{

        n=a[i]-j;

        if(n==strlen(w))

        {

                t=0;

                for(l=0;w[l];l++)

                {

                        if(s[l+j]==w[l])

                        {

                                t++;
```

```
                              }

                       }

                   if(t==strlen(w))

                {

                              found++;

               }

              }

             j=a[i]+1;

        }

        printf("word '%s' is occurred count=%d ",w,found);

}
```

OUTPUT:

Enter the string: I AM A BOY SREYASH IS A BOY

Enter word to be searched: BOY

word 'BOY' is occurred count=2

10. Read a string and rewrite it in the alphabetical order.

```c
#include<stdio.h>
int main(){
        char str[100],temp;
        int i,j;
        printf("Enter the string: ");
        gets(str);
        printf("%s in ascending order is: ",str);
```

```c
        for(i=0;str[i];i++){

        for(j=i+1;str[j];j++){

        if(str[j]<str[i])

        {

        temp=str[j];

        str[j]=str[i];

        str[i]=temp;

        }

        }

}

        printf("%s\n",str);

return 0;

}
```

OUTPUT:

Enter the string: DCBEA

DCBEA in ascending order is: ABCDE

11. Print the Words Ending with Letter S.

```c
#include <stdio.h>

#include <string.h>

char str[100];

int main()

{

   int i, t, j, len;
```

```c
    printf("Enter a string : ");

    scanf("%[^\n]s", str);

    len = strlen(str);

    str[len] = ' ';

    for (t = 0, i = 0; i < strlen(str); i++)

    {

        if ((str[i] == ' ') && (str[i - 1] == 's'))

        {

            for (j = t; j < i; j++)

                printf("%c", str[j]);

            t = i + 1;

            printf("\n");

        }

        else

        {

            if (str[i] == ' ')

            {

                t = i + 1;

            }

        }

    }

return 0;

}
```

OUTPUT:

Enter a string : lets go sreyash

lets

12. Delete All Repeated Words in the line of text.

```c
#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main ()

{

        char str[100], word[100], arr[10][30];

        int i = 0, j = 0, k = 0, len1 = 0, len2 = 0, l = 0;

        printf ("Enter the string\n");

        gets (str);


        // converting the string into 2D array

        for (i = 0; str[i] != '\0'; i++)

        {

                if (str[i] == ' ')

                {

                        arr[k][j] = '\0';

                        k ++;

                        j = 0;

                }

                else

                {
```

```c
                    arr[k][j] = str[i];

                    j ++;

            }

    }

    arr[k][j] = '\0';

    j = 0;

    for (i = 0; i < k; i++)

    {

    int present = 0;

            for (l = 1; l < k + 1; l++)

            {

                    if (arr[l][j] == '\0' || l == i)

                    {

                            continue;

                    }

                    if (strcmp (arr[i], arr[l]) == 0) {

                            arr[l][j] = '\0';

                            present = present + 1;

                    }

            }

    }

    j = 0;

    for (i = 0; i < k + 1; i++)

    {

            if (arr[i][j] == '\0')
```

```
                    continue;

            else

                    printf ("%s ", arr[i]);

        }

        printf ("\n");

        return 0;

}
```

OUTPUT:

Enter the string

hii hello hii

hii hello