

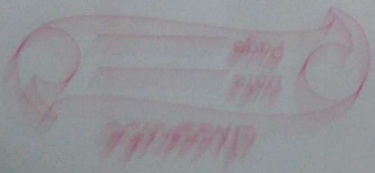
STRINGS

8/02/17

Collection of characters

```
char str[20] = "Delhi";
```

```
char str[20] = {'D', 'e', 'l', 'h', 'i', '\0'};
```



18 1a

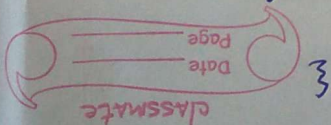
Q. WAP to input string character by character
(by using '%c' format specifier)

```
void main()
{
    char str[20], c;
    int i=0;
    printf("Enter string\n");
    do
    {
        scanf("%c", &str[i]);
        i++;
    } while (str[i-1] != '\n');

    str[i] = '\0';
    i = 0;
    while (str[i] != '\0')
    {
        printf("%c", str[i]);
        i++;
    }
    getch();
}
```

Q. WAP to input string using '%s' format specifier
or a single string.

```
void main()
{
    char str[20];
    printf("Enter string");
    scanf("%s", str);
    printf("%s", str);
    getch();
}
```



#include <string.h> - for turbo C

Q. WAP to input string and print it

```
void main()
{
    char str[20];
    printf("Enter string");
    gets(str);
    puts(str);

    getch();
}
```

Q. WAP to input string and print its length.

```
void main()
{
    char str[30];
    int i = 0;
    printf("Enter string");
    gets(str);

    while(str[i] != '\0')
    {
        i++;
    }

    printf("%d", i);
    getch();
}
```

a = strlen(str);

Q. WAP to input a string and copy the content of string to another string.

```
void main()
{
    char str[20], str1[20];
    int i = 0;
    printf("Enter string");
    gets(str);
    while (str[i] != '\0')
    {
        str1[i] = str[i];
        i++;
    }
    str1[i] = '\0';
    puts(str1);
    getch();
}
```

strcpy(str1, str)

Q. WAP to input a string and concatenate the strings into third string.

```
void main()
{
    char str[20], str1[20], str2[40];
    int i = 0, j = 0;
    printf("Enter 2 strings");
    gets(str);
    gets(str1);
    while (str[i] != '\0')
    {
        str2[i] = str[i];
        i++;
    }
    str2[i] = str1[j];
    j++;
    while (str1[j] != '\0')
    {
        str2[i] = str1[j];
        i++;
        j++;
    }
    str2[i] = '\0';
    puts(str2);
}
```

strcat(str2, str1);


```

while (str1[j] != '\0')
{
    str2[i] = str1[j];
    j++;
    i++;
}
str2[i] = '\0';
puts (str2);
getch();
}

```

Q. WAP to input two strings and compare them whether they are same or not.

```

void main()
{
    char str[30], str1[30];
    int a, b, flag = 0;
    gets (str);
    gets (str1);
    a = strlen (str);
    b = strlen (str1);
    if (a == b)
    {
        for (i = 0; str[i] != '\0'; i++)
        {
            if (str[i] != str1[i])
            {
                flag = 1;
                break;
            }
        }
        if (strlen (str) == strlen (str1))
        {
            if (strcmp (str, str1) == 0)
            {
                // strings are same
            }
        }
    }
}

```



```

1);
    if (flag == 0)
    { printf ("Equal");
    }
    else printf ("unequal");
else
    printf ("Unequal");
}
getch();
}

```

Q. WAP a program two strings, concatenate them into first string -

```

void main()
{
    char str[40], str1[20];
    int i, a;
    gets(str);
    gets(str1);
    a = strlen(str);
    for (i = 0; str1[i] != '\0'; i++)
    {
        str[a+i] = str1[i];
    }
    str[a+i] = '\0';
    puts(str);
    getch();
}

```


Q. WAP to input a string and convert all lower case character to upper case

```
void main()
{
    char str[20];
    int i;
    gets(str);
    for(i = 0; str[i] != '\0'; i++)
    {
        if (str[i] >= 97 && str[i] <= 122)
        {
            str[i] = str[i] - 32;
        }
    }
    puts(str);
    getch();
}
```

Q. WAP to input a string store it and print it in reverse order.

```
void main()
{
    char str[20], temp;
    int i; temp
    gets(str);
    for(i = 0; i < strlen(str)/2; i++)
    {
        temp = str[i];
        str[i] = str[strlen(str)-1-i];
        str[strlen(str)-1-i] = temp;
    }
}
```



```

temp = str[i];
str[i] = str[strlen(str)-1-i];
str[strlen(str)-i] = temp;
}

```

strrev(str)

```
puts(str);
```

```
getch();
```

```
}
```

Q. WAP to input a string and count how many of them vowels.

```
void main()
```

```
{
```

```
char str[20];
```

```
int i, c=0;
```

```
gets(str);
```

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

```
{
```

```
if (str[i]=='a' || str[i]=='e' || str[i]=='i' || str[i]=='o'
```

```
str[i]=='u' || str[i]=='A' || str[i]=='E' ||
```

```
str[i]=='I' || str[i]=='O' || str[i]=='U')
```

```
{
```

```
c++;
```

```
}
```

```
printf("%d", c);
```

```
getch();
```

```
}
```


Q. WAP to input a string and check whether it's a pallindrome or not.

```
void main()
```

```
{
    char str[30];
```

```
    int i, l, flag = 0;
```

```
    gets(str);
```

```
    l = strlen(str);
```

```
    for(i = 0; i < l/2; i++)
```

```
    {
        if(str[i] != str[l-i-1])
```

```
        flag = 1;
```

```
        break;
```

```
    }
```

```
    if(flag == 0)
```

```
        printf("Pallindrome");
```

```
    else
```

```
        printf("Not Pallindrome");
```

```
}
```

Q. WAP to input a string and count how many words are there in that string.

```
void main()
```

```
{
```

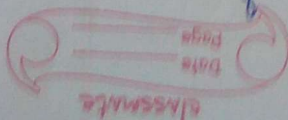
```
    char str[30];
```

```
    int i, c = 1;
```

```
    gets(str);
```

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

```
    {
        if(str[i] == ' ' && str[i+1] != ' ')
```




```

    c++;
}
printf("%d", c);
}

```

```

if (str[0] == ' ')
    c = 0;
else
    c = 1;
if (str[i] == ' ' && str[i+1] != ' ')

```

Q. WAP to input name and print its initials.

```

void main()
{
    char str[30];
    int i, j;
    printf("Enter string");
    gets(str);
    if (str[0] != '\0')
        printf("%d", str[0]);
    for (i = 1; str[i] != '\0'; i++)
    for (i = 0; str[i] != '\0'; i++)
    {
        if (str[i] == ' ' && str[i+1] != ' ')
            printf("%d", str[i+1]);
    }
}

```

```

strlen(str)
strupr(str)

```


Q. WAP to input string and count how many are consonants

```
void main()
{
    char str[20],
    int i, c=0;
    gets(str);
    strlwr(str);
    for (i=0; str[i]!='\0'; i++)
    {
        str[i] = tolower(str[i]);
        if (str[i]!='a' || str[i]!='e' || str[i]!='o' || str[i]!='i'
        str[i] = 'u')
        c++;
    }
    printf("%d", c);
    getch();
}

if (str[i]>=97 && str[i]<=122)
{
    if (str[i]!='a' && str[i]!='e' && str[i]!='o' && str[i]!='i' && str[i]!='u')
        str[i] = 'u';
    c++;
}

printf("%d", c);
getch();
```