

```

#include <stdio.h>
int main()
{
    //----- Using Break part 1
    for(i=0; i<10; i++)
    {
        if(i>5)
            break;
        printf("%2d",i);    // output: 0 1 2 3 4 5
    }

    //----- Using Break part 2
    for(i=0; i<10; i++)
    {
        printf("%2d",i);    // output: 0 1 2 3 4 5
        if(i>5)
            break;
    }

    //----- Using Continue Keyword
    // def. of Break:  exits from the current loop or go to the } (closing
    brace) of nearest loop
    // def. of Continue: Starts with the current loop or go to the { (Opening
    brace) of nearest loop

    //----- DIFFERENCE B2N BREAK AND CONTINUE
    for(i=0; i<10; i++)
    {
        if(i>=5)
            break;    // exits from line no 33 again if the value < 5
        printf("%2d",i);    // output: 0 1 2 3 4 (using break)
                           // (for the values starting from 5 to 10 the loop "never
runs")
    }

```

```
//-----  
for(i=0; i<10; i++)  
{  
    if(i>=5)  
        continue;           // starts from line no 38 again if the value < 5  
    printf("%2d",i);         // output: 0 1 2 3 4  
}  
// (for the values starting from 5 to 10 the "loop runs  
but no values will get printed")
```

//----- Using Continue part 2

```
for(i=0; i<10; i++)  
{  
    if(i==5)  
        continue;           // starts from line no 101 again if the value < 5  
    printf("%2d",i);         // output: 0 1 2 3 4 6 7 8 9  
}
```

//----- Using Break

```
for(i=0; i<10; i++)  
{  
    if(i==5)  
        break;  
    printf("%2d",i);         // output: 0 1 2 3 4  
}  
return 0;  
}
```

```
#include <stdio.h>
int main()
{
    int i, j, k, n;      // Consider i for each line, j for each Star and k for each
Space
    printf("Enter no of rows:\t");
    scanf("%d",&n);

    for(i=1;i<=n;i++)    // for each line .. ok but how many lines are there? n
lines
    {
        for(k=1;k<= ? ;k++) // will print spaces at each line
            printf(" ");

        for(j=1;j<= ?? ;j++) // will print * at each line
            printf("*");

        printf("\n");
    }
    return 0;
}
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1		n	i		FORMULA			n	n-i+1	i-1	FORMULA			n	2*i-1	n-i	FORMULA
2	n	i Row	j Star	k space	T1 Pattern Looks Like	n	i Rows	j Star	k space	T2 Pattern		n	i Rows	j Star	k space	T2 Pattern Looks Like	
3	4	1	1		*	4	1	4	0	****		4	1	1	3	*	
4		2	2		**		2	3	1	***			2	3	2	***	
5		3	3		***		3	2	2	**			3	5	1	*****	
6		4	4		****		4	1	3	*			4	7	0	*****	
7			i	n-i	FORMULA				n-i+1		FORMULA				2*(n-i)+1	i-1	FORMULA
8	n	i Row s	j Star	k space	T2 Pattern Looks Like	n	i Rows	j Star	k space	T1 Pattern Looks Like		n	i Rows	j Star	k space	T2 Pattern Looks Like	
9	4	1	1	3	*	4	1	4		****		4	1	7	0	*****	
10		2	2	2	**		2	3		***			2	5	1	*****	
11		3	3	1	***		3	2		**			3	3	2	***	
12		4	4	0	****		4	1		*			4	1	3	*	

//HCF AND LCM

//x= 4 and y=16

// Factors 4= 1 2 4 and 16= 1 2 4 8 16

// Common factors 1 2 4

// HCF 4

// HCF of 2, 3 HCF will be 1

// Multiples 4= 4 8 12 16 20 24 28 32 ... and 16= 16 32

// Common multiple 16 32...

// LCM of 4 and 16 will be 16

// LCM should exist in between of max(x,y) to x*y 16 to 64

// 2 and 3 LCM 3 to 6

// 2 and 3 HCF 2 to 1

// mn= min(x,y) e.g. 4

// every HCF should exist within 1 to minimum of the given two numbers

//

~~~~~  
~~`

mn= x>y ? y : x; // mn will contain the minimum of two numbers for HCF

for(i=mn; i>=1; i--)        //searching for the HCF between two numbers

{

  if(x%i==0    && y%i==0)

  {

    printf("HCF will be %d", i);

    break;

  }

}

//

~~~~~  
~~`

mx= x>y ? x : y; // mx will contain the maximum of two numbers for LCM

```

for(i=mx; i<=x*y; i++)    //searching for the LCM between two numbers
{
    if(i%x==0  && i%y==0)
    {
        printf("LCM will be %d", i);
        break;
    }
}
//

```

~~~~~  
 ~~~`

```

// finding out the Sum-of-digits
// input 348  => 3+4+8  output 15 (sum of each digits)

```

```

s=0;
scanf("%d",&n);
while(n!=0)
{
    r= n%10;
    n=n/10;
    s=s+r;
}
printf("sum of digit is %d", s);
//

```

~~~~~  
 ~~~`

```

// reverse a digit  i/p 348  o/p 843

```

```

s=0;
scanf("%d",&n);
while(n!=0)
{
    r= n%10;

```

```
    n=n/10;
    s=s*10+r;
}
printf("sum of digit is %d", s);
//
~~~~~
~~`
```