

* #include <stdio.h>
#include <conio.h>
int main()
{

 clrscr();
 printf("Hello ");
 printf("In C");
 return 0;

{}

* #include <stdio.h>
int main()

{

 printf("Hello ");
 return 0;

{}

2. #include <stdio.h>
 #include <conio.h>
 int main ()
 {
 clrscr();
 printf("1 Hello");
 printf("In 2 Hello");
 printf("In 3 Hello");
 printf("In 4 Hello");
 printf("In 5 Hello");
 return 0;
}

1. $9 \times 4 \div 2 + 12 = 30$
2. $6 \times 9 + 45 \div 5 = 60$
3. $123 \div 1 \times 2 = 63 \div 183$
4. $8 + 30 \times 7 \div 6 = 43$
5. $11 \times 11 \div 6 \times 12 = 24$
6. $8 \times 6 + 75 \times 3 = 273$
7. $70 - 55 + 63 - 40 = 38$
8. $10 \times 1 + 27 \times 12 = 334$
9. $8 \times 2 \div 6 \times 12 = 24$
10. $21 + 72 + 31 + 1 = 30$

```
#include <stdio.h>
#include <conio.h>
int main();
{
```

```
clascro ();
printf ("In %d + %d / (%d * %d) - %d = %d\n",
       10+20, (5*4)-1, 10+20, 5*4-1);
return 0;
```

* Variable

: It is a way to represent memory location through symbols so that it can be easily identified.

* Rules

- A variable can have alphabets, digits and underscore.
- A variable's name can start with the alphabet and underscore only. It can't start with a digit.
- No whitespace is allowed within the variable name.
- A variable name must not be any reserved word or keyword. e.g. if int, float, etc

Variable name should be meaningful and easy to understand.

```
#include <stdio.h>
#include <conio.h>
int main()
{
    int x=10, y= 20, z;
    clrscr();
    printf(" x = %d", x);
    printf(" y = %d", y);
    printf(" In x+y = %d", x+y);
    z = x+y;
    printf(" In x+y = %d ", z);
    getch();
}
```

* #include <stdio.h>
* #include <conio.h>
int main ()
{
 int a1=10, b1=20, c1=5, d1=4, e1=1,
 z;
 int a=10, b=30, c=5, d=4,
 e=1 = z1;
 clrscr();
 z = a+b/c * d - e;
 z1 = a1+b1/c1 * d1 - e1;

 printf("a+b/c*d-e=%d", z);
 printf("In a1+b1/c1*d1-e1=%d", z1);

 return 0;
}

* void main () = getch();
* main () = return 0;
* int main () = return 0;

```
#include <stdio.h>
#include <conio.h>
int main ()
{
    int x=10, y=20;
    clrscr();
    printf("In address of x=%u",
        &x);
    printf(" In value of x=%d ", x);
    printf("In address of y=%u",
        &y);
    printf("In value of y=%d ", y);
    return 0;
}
```

- %u = positive or negative value
- %d = positive value only
- Address is not change
- Value is change.

→ 11 → single line comment

1* → multi

line

comment

* /

```
# include < stdio.h >
# include < conio.h >
int main()
{
    printf("1 Hello");
    /* printf("2 Hello");
    printf("3 Hello");
    /* printf("4 Hello");
    printf("5 Hello");
    return 0;
```

2

```
* #include <stdio.h>
# include <conio.h>
int main ()
{
    int a=10, b=20, c;
    clrscr();
    printf (" a = %d ", a);
    printf (" b = %d ", b);
    printf (" In a+b = %d ", a+b);
    c=a+b;
    printf (" In a+b = %d ", c);
    return 0;
}
```

Convert

* 11 dollar to rupee

11 dollar = 75 rupee

* #include <stdio.h>

#include <conio.h>

int main()

{ int rupee , dollar=5;

class();

rupee = dollar * 75;

printf(" %d dollar = %d rupee ",

dollar , rupee);

return 0;

}

questions

1. c programme to calculate Area of circle.
2. c programme to calculate Area of square.
3. c programme to calculate Area of rectangle.
4. c programme to calculate Area of triangle.
5. c programme to convert days to year , month , week and day
6. $600 \rightarrow 1 \text{ year} , 7 \text{ month} , 0 \text{ weeks} , 0 \text{ days}$

data

12, 43, 54, 65, 55

type	datatype	format
number	int	specif- .0d

3.14, 2.1478, 5.25	Decimal	float	of f
--------------------	---------	-------	------

A, B, G, E, h	Alpha	char	of c
---------------	-------	------	------

sunct, pen, bike	word	string	of s
		char[]	

empty	void
-------	------

null

Operator

Type

+, -, *, /, %

Arithmatic operator

<, <=, >, >=, !=, ==

Relational operator

!, ||, !!, !

Logical operator

&, |, <<, >>, ~, ^

Bit wise operator

=, +=, -=, *=, /=, %=

Assignment operator

unary
operator

{ ++, -- }

Ternary
operatorTernary or
conditional operator

% → modulus

10 % 3 = 10 modulus 3

$$\begin{array}{r} 3 \\ \sqrt{10} \\ \hline \end{array}$$

$$\begin{array}{r} 9 \\ \hline 01 \end{array}$$

→ remainder

// conditional statement
// if
/*
if (expression) // if condition
{
 statement
}

#include <stdio.h>
#include <conio.h>
void main()
{
 clrscr();
 if (0)
 {
 printf("In inside if ");
 }
 getch();

```

★ #include <iostream.h>
# include <conio.h>
void main()
{
    clrscr();
    if(1)
    {
        printf("In It inside if");
    }
    else
    {
        printf("In It inside else");
    }
    getch();
}

```

16 i conditional statement

ii if

1*

if (expression) ii if condition

{

statement true part

}

else

{

}

statement false part

$$12 \cdot 1 \cdot 3 = 0$$

$$16 \cdot 1 \cdot 5 = 1$$

$$18 \cdot 1 \cdot 4 = 2$$

$$70 \cdot 1 \cdot 5 = 0$$

$$88 \cdot 1 \cdot 9 = 7$$

$$82 \cdot 1 \cdot 9 = 1$$

$$00 \cdot 1 \cdot 9 = 8$$

$$00 \cdot 1 \cdot n = n-1$$

↳ i) backslash and "Hacking"

(cmd)

(os)

A	B	A dd B	A B	A	!A
1	1	1	1	1	0
1	0	0	1	0	1
0	1	0	1	1	0
0	0	0	0	0	1

```
# include <stdio.h>
# include <conio.h>
void main()
{
    int a=100 , b=20;
    clrscr();
    if (a>b)
    {
        printf("In a=%d is greater", a);
    }
    else
    {
        printf("In b=%d is greater", b);
    }
    getch();
}
```

Questions

1. Enter two value find its smallest.
2. Enter One value find its positive or negative.
3. Enter One value and find its divisible by 5 or not.
4. Enter One value and find its divisible by 5 and 3 or not.
5. Enter two value and find its multiplication is divisible by 3 and 7 or not.
6. Enter two value and find multiplication is divisible by its addition or not.

* Enter Three value and find largest

```

★ #include <stdio.h>
# include <conio.h>
void main()
{
    char ch;
    clrscr();
    printf("Enter ch= ");
    scanf("%c", &ch);
    printf("In ch=%c ", ch);
    getch();
}

```

→ ASCII = American Standard code for Information Interchanging

- 11 A - 65 to z - 90

// difference 32

11 a - 97 to z = 122

Questions

1. Enter a character vowel (a, e, i, o, u) or constant?
2. convert case → a - A

(+1)

(-1)

++ - increment -- - decrement

$c = c + 1$ $c = c - 1$

inc $(++)$

dec $(--)$

post

pre

post

pre

diff: "++c" "++c"

"c--"

--c

post

pre

assign



operat

operat



assign

* 11 post

```
#include <iostream.h>
#include <conio.h>
void main()
{
    int a = 10;
    clrscr();
    printf(" a = %d ", a++); 10
    printf(" In a = %d ", a++); 11
    printf(" In a = %d ", a); 12
    getch();
}
```

* 11 pre

```
#include <iostream.h>
#include <conio.h>
void main()
{
    int a = 10;
    clrscr();
    printf(" a = %d ", ++a); 11
    printf(" In a = %d ", ++a); 12
    printf(" In a = %d ", a); 12
    getch();
}
```

$$cl = 23$$

- | | | |
|-------------|------|-------------------|
| 1. $a + cl$ | = 24 | ($a - cl$) = 49 |
| 2. $a + +$ | = 24 | 2. $a - -$ = 49 |
| 3. $cl + +$ | = 25 | 3. $--a$ = 47 |
| 4. $+ + a$ | = 25 | 4. $cl --$ = 47 |
| 5. $cl + +$ | = 25 | 5. $--a$ = 45 |
| 6. $+ + cl$ | = 29 | 6. $a - -$ = 45 |
| 7. $cl + +$ | = 29 | 7. $--a$ = 43 |
| 8. $+ + cl$ | = 31 | 8. $a - -$ = 43 |
| 9. $cl + +$ | = 31 | 9. $--a$ = 41 |
| 10. a | = 32 | 10. a = 41 |

$$a = 77$$

- | | | |
|-------------|------|----|
| 1. $a - -$ | = 76 | 77 |
| 2. $+ + a$ | = 77 | 77 |
| 3. $cl - -$ | = 76 | 77 |
| 4. $a + +$ | = 77 | 76 |
| 5. $--a$ | = 76 | 76 |
| 6. $a + +$ | = 77 | 76 |
| 7. $a - -$ | = 76 | 77 |
| 8. $+ + cl$ | = 76 | 77 |
| 9. $cl - -$ | = 78 | 77 |
| 10. a | = 75 | 76 |

* #include <stdio.h>
#include <conio.h>
void main ()
{
 int a=10;
 clrscr();
 printf("Hello"), printf("World")
 , printf("Nico");
 printf("In %d %d %d %d"
 , a++, a++);
 getch();
}

II continue statement / loop
II for while do while
II for
1*
for(exp=1; exp=2; exp=3)
{
 statement
}

exp 1 = initial : i=1 . eval only once
exp 2 = condition : 1 <= 10 , every before
exp 3 = inc / dec : i++, every before
enter
exit

* #include < stdio.h > // header file
#include < conio.h > // header file
void main ()
{
 int i; // i is local variable
 clrscr();
 for (i=1; i<=10; i++)
 {
 printf("i=%d Hello ", i);
 getch();
 }
}

initial condition	statement	include
i=1	i<=10	
	1 <= 10	Hello 1
	2 <= 10	Hello 2
	3 <= 10	Hello 3
	4 <= 10	Hello 4
	5 <= 10	Hello 5
	6 <= 10	Hello 6
	7 <= 10	Hello 7
	8 <= 10	Hello 8
	9 <= 10	Hello 9
	10 <= 10	Hello 10
	11 <= 10	false X

*/

```

★ #include <stdio.h>
# include <conio.h>
void main()
{
    int i=1; if (i<10)
    clrscr(); else
    for (j;j<10)
    {
        if (i==10)
            printf("In %d", i++);
        else
    }
}

```

Bacak;

getch();

1	0110H
2	0110H
3	0110H
4	0110H
5	0110H
6	0110H
7	0110H
8	0110H
9	0110H
00	0110H
11	0110H
X	0110H

★ notepad

```

★ #include <stdio.h>
int main()
{
    int i, n;
    for (i=1; i<=4; i++) { // show in row
        for (n=1; n<=43; n++) { // show in column
            printf("*");
        }
        printf("\n");
    }
    return 0;
}

```

i = 1	i <= 4	n = 1	1 <= 3	n++
1	1 <= 4	1	1 <= 3	* * *	2
			2 <= 3	* * *	3
			3 <= 3		4
			4 <= 3	(X)	
2	2 <= 4	1	1 <= 4		!++
			2 <= 4		2
			3 <= 4		3
			4 <= 4		4
			5 <= 4		5
			5 <= 4	(X)	

* int

```
# include <stdio.h>
int main ()
{
    int i, n;
    for (i=1; i<=5; i++)
        for (n=1; n<=i; n++)
            printf ("%d * ", i);
        printf ("\n");
    return 0;
}
```

```
#include <stdio.h>
int main ()
{
    int i, j=1, n, sp=10;
    for (j=1; j<=5; j++)
    {
        for (n=1; n<=sp; n++)
            printf("    ");
        for (n=1; n<=j; n++)
            printf("* ");
        printf("\n");
        sp--;
    }
    return 0;
}
```

★

```
#include <stdio.h>
int main ()
{
    int i;
    for (i=1; i<=10; i++)
        printf("In %d", i);
    i=1;
    while (i<=10)
        printf("In %d", i);
        i++;
    }
    return 0;
}
```

★ #include<stdio.h>

int main()

{

 int i = 1;

 while (1) → [always true]

 for (; ;)

{

 if (i == 26)

 printf("In %c - %c",

 i + 64, i + 64 + 33);

 else

{

 Break;

return 0;

{

Number

System

Decimal

0 - 9

Binary

0 or 1 ✓ computer use

Octal

0 - 7

Hexa decimal

0 - 15

10 - A

11 - B

12 - C

13 - D

14 - E

15 - F

Decimal

Binary

1010

10

612

0

312

0

212

1

112

0

1

 $(1010)_{10} \rightarrow (1100)_2$

A) 35

2512	126	1
1212	6	0
612	3	0
312	1	1
112	-	1

$$\begin{array}{l} 8400 \\ 0801 \\ \downarrow \\ (25)_{10} \\ (11001)_2 \end{array}$$

A) 36

3612	18	0
1812	9	0
912	4	1
412	2	0
212	1	0
112	-	1

$$\begin{array}{l} 9 - 1 \\ 8 - 1 \\ \downarrow \\ (36)_{10} \\ (100100)_2 \end{array}$$

A) 49

4912	24	1
2412	12	0
1212	6	0
612	3	0
312	1	1
112	-	1

$$\begin{array}{l} 9 - 1 \\ 8 - 1 \\ \downarrow \\ (49)_{10} \\ (110001)_2 \end{array}$$

~~ESTAO~~

$2^0 - 1$

$2^1 - 2$

$2^2 - 4$

$2^3 - 8$

$2^4 - 16$

$2^5 - 32$

$2^6 - 64$

$2^7 - 128$

$2^8 - 256$

$2^9 - 512$

$2^{10} - 1024$

$2^{11} - 2048$

↓
cooboo

↓
(100011)

* #include <stdio.h>

```
int main()
{
    int i=1, n, f, sp=10;
    while(i<=5)
    {
        n = i;
        while(n>=sp)
        {
            printf("   ");
            n--;
        }
        f = 1;
        while(f<=i)
        {
            printf("*");
            f++;
        }
        printf("\n");
        i++;
        sp -= i;
    }
    return 0;
}
```

```
#include <stdio.h>
int main()
{
    int i=1, n, f, sp=10; i
    do
    {
        n = 1; j
        do
        {
            printf("    "); i
            n++; m
        } while (n <= sp); l
        f = 1; j
        do
        {
            printf(" * "); i
            f++; m
        } while (f <= i); l
        printf("\n"); i
        i++; m
    } while (i <= sp); l
    return 0;
}
```

→ ★ Array

↳ Homogeneous data

collection of data

which has all same datatype

* II Homogeneous data

#include <stdio.h>

int main()

{

 int a[5], i;

 for (i=0; i<5; i++) // insert

{

 printf(" Enter a[%d] : ", i);

 scanf("%d", &a[i]);

}

 for (i=0; i<5; i++) // print

{

 printf(" In a[%d] : %d ", i, a[i]);

}

return 0;

}

a[0] : 12

a[1] : 34

a[2] : 54

a[3] : 65

a[4] : 77

index start from: 0.

Scanned with CamScanner

Task:

1. Odd position element total
2. Even position element total
3. Odd element total
4. Even Element total
5. All Element total

2D Array

```
#include <stdio.h>
int main()
{
    int i, n, a[3][3];
    for(i=0; i<3; i++)
        for(n=0; n<3; n++)
            printf("Enter a[%d][%d], ", i, n);
            scanf("%d", &a[i][n]);
    printf("In matrix a In In ");
    for(i=0; i<3; i++)
    {
        for(n=0; n<3; n++)
            printf(" %d ", a[i][n]);
        printf("\n");
    }
}
```

1 2 3
4 5 6
7 8 9

Total dicmingle : $1+5+9 = 15$

zepper Tatimgle : $2+3+6 = 11$

lower Tatimgle : $4+7+8 = 19$

* Storing

1* .d g J fc

→ Street → class of cheer
stair[10] : "s k i l l o"
0 1 2 3 4 5

#include <stdio.h>

int main()

{

cheer stair[10];

printf("Enter Stairing: ");

scanf("%s", stair);

printf("In Stairing = %s ", stair);

}

^A

^ → correct

```
#include <stdio.h>
int main()
{
    char str[10];
    printf("Enter string: ");
    scanf("%[^\\n]", str);
    printf("String = %s", str);
}
```

```
#include <stdio.h>
int main()
{
    char str[10];
    int i;
    printf("Enter string : ");
    gets(str);
    for(i=0; str[i] != '\0'; i++);
    printf("Length = %d ", i);
```

}

★

5 6 7 8 9 10
0 1 2 3 4 5

★/

Output :

Enter string : skill

length: 5

function

string length
1) string

#include <stdio.h>

#include <string.h>

int main()

{

char str[100];

int len;

printf("Enter string: ");

gets(str);

len = strlen(str);

printf("Length of string = %d ", len);

return 0;

}

Output :-

z edhgvj

```
#include <stdio.h>
#include <string.h>
int main()
{
    char str[100];
    int len;
    printf("Enter string: ");
    gets(str);
    puts(str);
    len = strlen(str);
    printf("String length = %.d ", len);
    strrev(str);
    puts(str);
    strpbrk(str);
    puts(str);
    strlwr(str);
    puts(str);
}
```

```
#include <stdio.h>
#include <string.h>
int main()
{
    char str1[100], str2[100];
    int len;
    printf("Enter string 1: ");
    gets(str1);
    printf("Enter string 2: ");
    gets(str2);
    strcpy(str2, str1);
    printf("In strcmp = %d", strcmp(str1, str2));
    printf("In strcmpi = %d In", strcmpi(str1, str2));
    strcasecmp(str2, "");
    strcasecmp(str2, str1);
    puts(str2);
    printf("In strlen = %d In", strlen(str2));
}
```

strlen(str) = length

strncpy(str), servers

strcmp(str) = capital

strlwr(str) = small

strcpy(str, str1) = copy

strcmp(str, str1), compare sensitive
o. same

strcmpi(str, str1) :

occurrences

uppercase & lowercase

multiple occurrences - ; comparison

substrings

* // function ; revise

// syntax : function definition

// return-type <function name>
(parameter list ...)

// {

// statements

// }

// void sum()

// {

a+b

// }

// function calling

// <function name>(); // call

// void sum(); → function declaration

// return

- parameter

#

11 NRNP

```
#include <stdio.h>
void myfun()
{
    printf("In Hi, you are inside
myfun...");}

int main()
{
    printf("In Hi, you are before
myfun()");  

    printf("In Hi, you are after
main()");}
```

AIRNP.

```
#include <stdio.h>
```

```
void dosum ()
```

```
{
```

```
int a, b;
```

```
printf("Enter a = ");
```

```
scanf("%d", &a);
```

```
printf(" Enter b = ");
```

```
scanf("%d", &b);
```

```
printf(" Int a+b = %d ", a+b);
```

```
}
```

```
int main ()
```

```
{
```

```
dosum();
```

```
}
```

WPNR

PAGE NO.
DATE:

```
#include <stdio.h>
void dosum(int a, int b)
{
    printf("a+b = %d", a+b);
}
int main()
{
    int a,b;
    printf("enter a = ");
    scanf("%d", &a);
    printf("enter b = ");
    scanf("%d", &b);
    dosum(a,b);
}
```

NPWR

```
# include <stdio.h>
```

```
int dosum()
```

{

```
    int a, b;
```

```
    printf("Enter a = ");
```

```
    scanf("%d", &a);
```

```
    printf("Enter b = ");
```

```
    scanf("%d", &b);
```

}

```
int main()
```

{

```
    int sum;
```

```
    sum = dosum();
```

```
    printf("In a+b=%d", sum);
```

}

dosum()

→ return multiple value return is ↴

→ return single value return is ↴

APNR

```
#include <stdio.h>
```

```
int m dasum(int x, int y).
```

{

```
return x+y;
```

}

```
int main()
```

{

```
int a, b;
```

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

```
scanf("%d", &a);
```

```
printf(" Enter b : ");
```

```
scanf("%d", &b);
```

```
printf(" a+b = %d ", dasum(a,  
b));
```

}

→ Variable same datatype

```
#include <stdio.h>
```

```
int main()
```

{

```
    int x;
```

```
    int myfun(int, int) — function
```

```
declaration
```

```
    x = myfun(10, 20);
```

```
    printf("In inside main... %d", x);
```

}

```
int myfun(int n, int m) — function
```

```
definition
```

{

```
    printf("In inside myfun... ");
```

```
    return n+m;
```

}

out put

inside myfun...

inside main ... 30

11 Recursion

```
#include <stdio.h>
int myfun(int n)
{
    printf(" winding = %d ", n);
    if (n > 3)
        myfun(n + 1);
    printf(" unwinding = %d ", n);
}
```

```
int main()
```

```
printf("In inside main --- ");
MyFun(2);
```

3

Output:-

inside main ---

winding = 1

winding = 2

winding = 3

un-winding = 3

un-winding = 2

un-winding = 1

```
#include <iostream.h>
int fact (int n)
```

```
if (n == 1 || n == 0)
```

```
return 1;
```

```
else
```

```
n * fact (n - 1);
```

```
int main ()
```

```
{
```

```
int factorial, n = 5;
```

```
factorial = fact (n);
```

```
printf ("The factorial of %d is %d", n, factorial);
```

Output :-

$n = 5$

main () -> fact (5)	5	$5 \times \text{fact}(5-1)$
(4)	4	$4 \times \text{fact}(4-1)$
(3)	3	$3 \times \text{fact}(3-1)$
(2)	2	$2 \times \text{fact}(2-1)$
(1)	1	

Output :- 120

* pointer

```
#include <stdio.h>
```

```
int main()
{
```

```
    int a = 10, *p;
    p = &a;
```

```
    printf("In value of a = %d", a);
```

```
    printf("In address of a = %u", &a);
```

```
    printf("In value of using p = %d",
           *p);
```

```
    printf("In address of using p =
           %u", p);
```

```
    printf(" address of p = %u", &p);
```

(E-7) + 20 x 2

(E-7) + 20 x 3

(E-7) + 20 x 4

(E-7) + 20 x 5

(E-7)

(7) true (1) false

(7) (1)

(7) (8)

(7) (9)

(7)

2 pointer

#include <stdio.h>

int main()

{
 int a=10, *p, **p2;
 p=&a;
 p2=&p;

printf("In value of a: %d", a);

printf("In address of a: %u", &a);

printf("In value of a using p=%d", *p);

printf("In address of a using p=%u", p);

printf("In address of p=%u", p);

printf("In value of a using p2: %d", **p2);

printf("In address of a using p2: %u", *p2);

printf("address of a using p2: %u", p2);

printf("address of p2: %u", p);

```
#include <stdio.h>
int main()
{
    int a[5] = {23, 43, 54, 51, 61};
    int i, *ptr = NULL;
    ptr = a;
    printf("In address: %d Value In %d\n");
    printf("In %d Value In %d\n");
    for(i=0; i<5; i++)
    {
        printf("In %d + %d", a[i], a[i]);
        printf("In %d + %d", *ptr+i);
        printf("In %d + %d", *(ptr+i));
    }
}
```

Ternary operator.

```
#include <stdio.h>
int main()
```

```
{ int a=100, b=200;
```

```
if (a>b) ? printf("In a(%d) is greater", a) : printf("In b(%d) is greater", b);
```

// condition ? true : false

5.

Output of Program

Program

: a 9999

Output of Program

Program

: b 9999

Output of Program

Program

: a 9999

Output of Program

Program

switch

#include <stdio.h>

int main()

int a=100, b=20, ch='a';

if (a>b)? printf("In a (i.e. a) is greater", a);
printf("In b (i.e. b) is greater", b);

switch(ch)

case 1:

printf("In Monday");

break;

case 2:

printf("In Tuesday");
break;

case 3:

printf("In Wednesday");
break;

case 4:

printf("In Thursday");
break;

(case 5%
back;

printf("In Friday");

default:

printf("In Waeng choice III");
back;

}

{

label: abap

at 6 8 F 2 2 4 8 5 . t

★

27-7-112

```
#include <stdio.h> //using  
int main()  
{  
    int i = 1; //initialization  
    label:  
    printf("%d\n", i);  
    i++;  
  
    if(i == 10)  
        goto label;  
}
```

OUTPUT :-

1 2 3 4 5 6 7 8 9 10

A

```
#include <stdio.h>
int main ()
{
    int i = 1;
    printf ("1n %.2d", 10);
    printf ("1n %.3d", 10);
    printf ("1n %.4d", 10);
    printf ("1n %.5.2f", 10.0 / 3.0);
```

3

output :- 10

10

10

output :- 3.33

* #include <string.h>

#include <stdio.h>

struct student

{

int rollno;

char name[10];

int m1, m2, m3, total;

float per;

} ;

int main()

{

int i;

struct student st[5];

printf(" Enter Records of 2 students");

for (i=0; i<2; i++)

{

printf(" Enter Rollno: ");

scanf("%d", &st[i].rollno);

printf(" Enter name: ");

scanf("%s", &st[i].name);

printf(" Enter m1 = ");

scanf("%d", &st[i].m1);

printf(" Enter m2 = ");

scanf("%d", &st[i].m2);

printf(" Enter m3 = ");

scanf("%d", &st[i].m3);

}

```
printf("In student information  
list : ");  
for(i=0; i<2; i++)  
{  
    str[i].total = str[i].m1 + str[i].  
                  m2 + str[i].m3;  
    str[i].per = str[i].total / 3;  
  
    printf("In Rollno : %d In name  
%s", str[i].rollno, str[i].name);  
    printf("In total : %d In  
percentage = %.2f ",  
          str[i].total,  
          str[i].per);  
}
```

```
return 0;
```

```
}
```

File:

```
#include <stdio.h>
int main()
```

```
FILE *fp;
```

```
fp = fopen ("c:/users/swapnil/desktop  
/skill.txt", "wt");
```

```
fp = fopen ("skill-1.txt", "wt");
```

```
fprintf(fp, "welcome to skill guide the  
programming lab 1n")
```

```
fputs ("you are entering to learn  
language now-1n", fp);
```

```
fclose(fp);
```

```
}
```

r

opens a text file in read mode

w

opens a text file in write mode

a

opens a text file in append mode

```
*#include <stdio.h>
int main()
{
    FILE *fp;
    char buff[255];
    fp = fopen ("skill.txt", "r");
    while (fscanf(fp, "%s", buff) != EOF)
    {
        printf("%s", buff);
    }
    fclose (fp);
    // printf("In In %s", buff);
}
```