

## LIST OF PRACTICALS

1. Program to print a content using printf & puts.
2. Program to use different data types (int, float, char)
3. Program to use arithmetic operators
4. Program to use logical operators
5. Program to use relational operators
6. Program to use increment and decrement operators
7. Program to use conditional statements: if-else, If else ladder
8. Program to use for loop, nested for loop
9. Program to use while loop, do-while loop
10. Program to use switch ( break & continue)
11. Program to implement and use functions
12. Program to use arrays
13. Program to use structures

### Practical Number 1

Practical name- Program to print a content using printf & puts.

#### Practical code:

// Program to print a content using printf

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

```
int main()
```

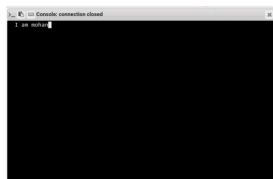
```
{
```

```
    printf("I am mohan");
```

```
    return 0;
```

```
}
```

#### Output:



// Program to print a content using puts

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

```
int main()
```

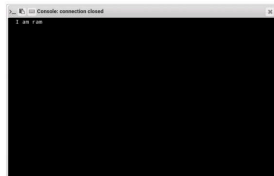
```
{
```

```
    puts("I am ram");
```

```
    return 0;
```

```
}
```

### Output:



### **Practical Number 2**

Practical name: Program to use different data types (int, char, float)

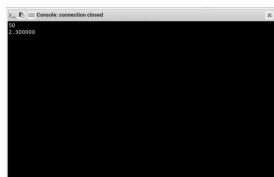
#### Practical code

```
include <stdio.h>
int main()
{
    int a=50;
    float b=2.30;
    char c='c';

    printf("%d\n",a);
    printf("%2f\n",b);
    printf("%c\n",c);

}
```

Output



### **Practical Number 3**

Practical name: Program to use arithmetic operators

#### Practical code

```
#include <stdio.h>
int main()
{
    int a,b,c;
    printf("Enter the value of a and b\n");
    scanf("%d%d", &a,&b);
```

```

c = a+b;
printf("a+b = %d \n",c);
c = a-b;
printf("a-b = %d \n",c);
c = a*b;
printf("a*b = %d \n",c);
c = a/b;
printf("a/b = %d \n",c);
c = a%b;
printf("Remainder when a divided by b = %d \n",c);
return 0 ; }

```

## Output

```

>_ _ _ _ _ Console: connection closed (Running: 17 seg)
Enter the value of a and b
a:
b:
a+b = 14
a-b = 2
a*b = 48
a/b = 3
a/b = 3      Remainder when a divided by b = 2

```

## Practical Number 4

Practical name: Program to use logical operators

### Practical code

```

#include <stdio.h>
int main()
{
    int a = 5, b = 5, c = 10, result;

    result = (a == b) && (c > b);
    printf("(a == b) && (c > b) is %d \n", result);

    result = (a == b) && (c < b);
    printf("(a == b) && (c < b) is %d \n", result);

    result = (a == b) || (c < b);
    printf("(a == b) || (c < b) is %d \n", result);

    result = (a != b) || (c < b);
    printf("(a != b) || (c < b) is %d \n", result);

    result = !(a != b);
    printf("(a != b) is %d \n", result);
}

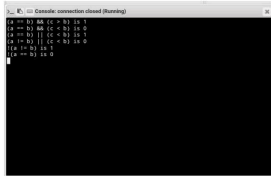
```

```

    result = !(a == b);
    printf("!(a == b) is %d \n", result);

    return 0;
}
Output

```



### Practical Number 5

Practical name: Program to use relational operators

#### Practical code

```

#include <stdio.h>
int main()
{
    int a = 5, b = 5, c = 10;

    printf("%d == %d is %d \n", a, b, a == b);
    printf("%d == %d is %d \n", a, c, a == c);
    printf("%d > %d is %d \n", a, b, a > b);
    printf("%d > %d is %d \n", a, c, a > c);
    printf("%d < %d is %d \n", a, b, a < b);
    printf("%d < %d is %d \n", a, c, a < c);
    printf("%d != %d is %d \n", a, b, a != b);
    printf("%d != %d is %d \n", a, c, a != c);
    printf("%d >= %d is %d \n", a, b, a >= b);
    printf("%d >= %d is %d \n", a, c, a >= c);
    printf("%d <= %d is %d \n", a, b, a <= b);
    printf("%d <= %d is %d \n", a, c, a <= c);

    return 0;
}
Output

```

```
> Console: connection closed
5 == 5 is 1
5 == 10 is 0
5 > 5 is 0
5 > 10 is 0
5 < 5 is 0
5 < 10 is 1
5 != 5 is 0
5 != 10 is 1
5 >= 5 is 1
5 >= 10 is 0
5 <= 5 is 1
5 <= 10 is 1
```

## Practical Number 6

Practical name: Program to use increment and decrement operators

### Practical code

```
#include <stdio.h>
int main()
{
    int a = 25, b = 30;
    float c = 15.5, d = 20.5;

    printf("a=25\b b=30\tc=25.5\td=10.5\n");
    printf("++a = %d \n", ++a);
    printf("--b = %d \n", --b);
    printf("++c = %f \n", ++c);
    printf("--d = %f \n", --d);

    return 0;
}
```

### Output

```
> Console: connection closed (Running)
a=25\b b=30\tc=25.5\td=10.5
++a = 26
--b = 29
++c = 16.500000
--d = 19.500000
```

## Practical Number 7

Practical name: Program to use conditional statements: if-else, If else ladder

### Practical code

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

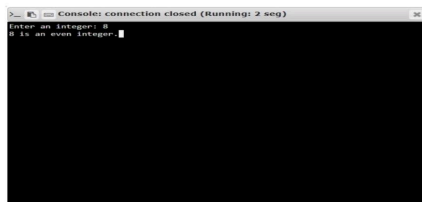
```

{
    int number;
    printf("Enter an integer: ");
    scanf("%d", &number);
    if (number%2 == 0)
    {
        printf("%d is an even integer.",number);
    }
    else
    {
        printf("%d is an odd integer.",number);
    }

    return 0;
}

```

Output



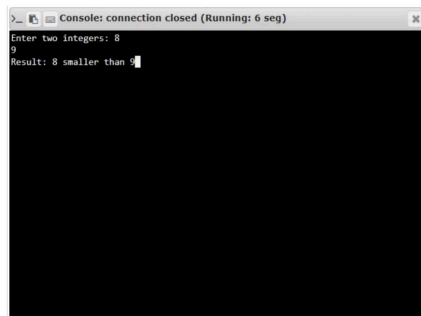
```

#include <stdio.h>
int main()
{
    int number1, number2;
    printf("Enter two integers: ");
    scanf("%d %d", &number1, &number2);
    if(number1 == number2)
    {
        printf("Result: %d equals to %d",number1,number2);
    }
    else if (number1 > number2)
    {
        printf("Result: %d greater than%d", number1, number2);
    }
    else
    {

```

```
    printf("Result: %d smaller than %d",number1, number2);  
}  
return 0;  
}
```

### Output



## Practical Number 8

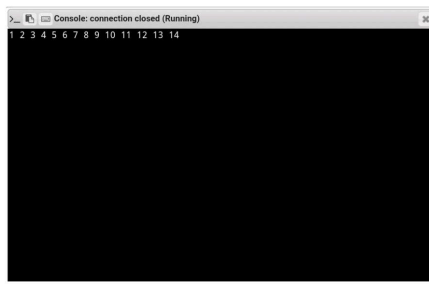
Practical name: Program to use for loop, nested for loop

### Practical code

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

```
int main()  
{  
    int i;  
  
    for (i = 1; i < 15; ++i)  
    {  
        printf("%d ", i);  
    }  
    return 0;  
}
```

### Output

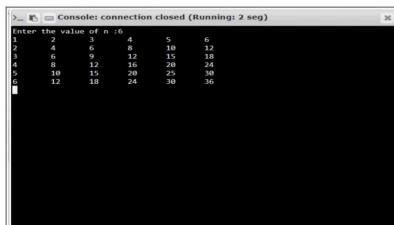


```

#include <stdio.h>
int main()
{
    int n;
    printf("Enter the value of n :");
    scanf("%d",&n);
    for(int i=1;i<=n;i++)
    {
        for(int j=1;j<=n;j++)
        {
            printf("%d\t",(i*j));
        }
        printf("\n");
    }
    return 0;
}

```

Output



### Practical Number 9

Practical name: Program to use while loop, do-while loop

#### Practical code

```

#include <stdio.h>
int main()
{
    int i = 1;
    while (i <= 20)
    {
        printf("%d\n", i);
        ++i;
    }
    return 0;
}

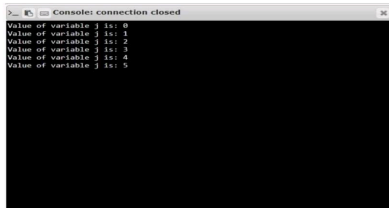
```

Output



A screenshot of a console window titled "Console: connection closed (Running)". The window displays a vertical list of numbers from 1 to 20, with each number on a new line. The numbers are aligned to the left of the window.

```
#include <stdio.h>
int main()
{
    int j=0;
    do
    {
        printf("Value of variable j is: %d\n", j);
        j++;
    }
    while (j<=5);
    return 0;
}
Output
```

A screenshot of a console window titled "Console: connection closed". The window displays the output of the first program, which is a vertical list of numbers from 1 to 5, each preceded by the text "Value of variable j is: ". The output is aligned to the left of the window.

## Practical Number 10

Practical name: Program to use switch ( break & continue)

### Practical code

```
#include <stdio.h>
int main()
{
    char n;
    int a,b;
    printf("Choose the operation\n");
    printf("Addition= +\n");
    printf("Subtraction= -\n");
    printf("Multiplication= *\n");
```

```

scanf("%c",&n);
printf("Enter the number=\n",a,b);
scanf("%d%d",&a,&b);
switch(n)
{
    case '+':
        //Addition
        {
            printf("Sum of two numbers is:\n");
            printf("%d and %d is=%d",a,b,a+b);
            break;
        }
    case '-':
        //Subtraction
        {
            printf("Difference of two numbers is:\n");
            printf("%d and %d is=%d",a,b,a-b);
            break;
        }
    case '*':
        //Multiplication
        {
            printf("Product of two numbers is:\n");
            printf("%d and %d is=%d",a,b,a*b);
            break;
        }
}
}

```

## Output

```

>> Console: connection closed (Running: 9 srg)
Enter the operation
Addition: +
Subtraction: -
Multiplication: *
Enter the number-
8
7
Difference of two numbers is:
8 and 7 is=1

```

## Practical Number 11

Practical name: Program to implement and use functions

### Practical code

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

```

int addition(int num1, int num2)
{
    int sum;
    sum = num1+num2;
    return sum;
}
int main()
{
    int var1, var2;
    printf("Enter number 1: ");
    scanf("%d",&var1);
    printf("Enter number 2: ");
    scanf("%d",&var2);
    int res = addition(var1, var2);
    printf ("Output: %d", res);
    return 0;
}
Output

```

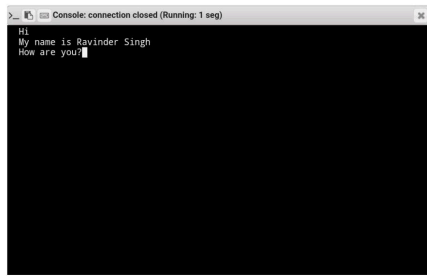


```

#include <stdio.h>
void introduction()
{
    printf("Hi\n");
    printf("My name is Ravinder Singh \n");
    printf("How are you?");
}
int main()
{
    introduction();
    return 0;
}

Output

```



```
>_ _> Console: connection closed (Running: 1 seg)
HI
My name is Ravinder Singh
How are you?
```

## Practical Number 12

Practical name: Program to use arrays

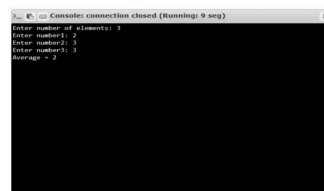
### Practical code

```
#include <stdio.h>
int main()
{
    int marks[10], i, n, sum = 0, average;
    printf("Enter number of elements: ");
    scanf("%d", &n);
    for(i=0; i<n; ++i)
    {
        printf("Enter number%d: ", i+1);
        scanf("%d", &marks[i]);
        sum += marks[i];
    }

    average = sum/n;
    printf("Average = %d", average);

    return 0;
}
```

Output



```
>_ _> Console: connection closed (Running: 9 seg)
Enter number of elements: 3
Enter number1: 2
Enter number2: 5
Enter number3: 3
Average = 2
```

## Practical Number 13

Practical name: Program to use structures

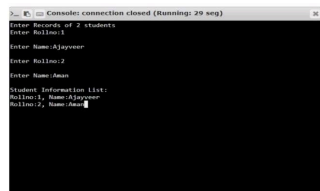
### Practical code

```

#include<stdio.h>
#include <string.h>
struct student
{
int rollno;
char name[10];
};
int main()
{
int i;
struct student st[2];
printf("Enter Records of 2 students");
for(i=0;i<2;i++)
{
printf("\nEnter Rollno:");
scanf("%d",&st[i].rollno);
printf("\nEnter Name:");
scanf("%s",&st[i].name);
}
printf("\nStudent Information List:");
for(i=0;i<2;i++)
{
printf("\nRollno:%d, Name:%s",st[i].rollno,st[i].name);
}
return 0;
}

```

## Output



```

C:\> Console: connection closed (Running: 29 srg)
Enter Records of 2 students
Enter Rollno:1
Enter Name:Ajayveer
Enter Rollno:2
Enter Name:Aman
Student Information List:
Rollno:1, Name:Ajayveer
Rollno:2, Name:Aman

```