

# Funksiyalar bilan ishlash

## (user-defined functions)

# Reja:

- Funksiyani e'lon qilish
- Funksiyaga murojaat qilish
- Funksiya parametrlari
- Default qiymatli parametrlar
- Amaliy mashqlar

# Funksiya

**Funksiya** - ma'lum bir vazifani bajaruvchi, qandaydir nomga ega, bir yoki bir nechta qiymatni qabul qiluvchi, ishni tugatganidan keyin esa asosiy dasturga biror yoki bir necha natija qiymatlarini qaytaruvchi qism dastur.

# Funksiyani e'lon qilish

# Syntax

```
returnType functionName (parameter1, parameter2,...) {  
    // function body  
}
```

# Example

```
int add (int a, int b) {  
    return (a + b);  
}
```

Bunda:

- 1) Funksiya nomi: **add**
- 2) Funksiyaning qaytaruvchi qiymat turi: **int**
- 3) Parametrlar: **int** turidagi **a** va **b**
- 4) Funksiya tanasi: **{}** qavslar ichida yozilgan kod



# Funksiyaga murojaat qilish

```
#include <iostream>

using namespace std;

// declaring a function
int add(int a, int b) {
    return (a + b);
}

int main() {

    int sum;

    // calling the function and storing
    // the returned value in sum
    sum = add(100, 78);

    cout << "100 + 78 = " << sum << endl;

    return 0;
}
```

```
#include<iostream>
```

```
int add(int a, int b) {  
    return (a + b);  
}
```

```
int main() {  
    int sum;  
    sum = add(100, 78);  
    ... ..  
}
```

function  
call

# Output

```
100 + 78 = 178
```

# Function prototype

# Syntax

```
returnType functionName(dataType1, dataType2, ...);
```

# Example

```
// function prototype  
int add(int, int);
```

```
#include <iostream>

using namespace std;

// function prototype
int add(int, int);

int main() {
    int sum;

    // calling the function and storing
    // the returned value in sum
    sum = add(100, 78);

    cout << "100 + 78 = " << sum << endl;

    return 0;
}

// function definition
int add(int a, int b) {
    return (a + b);
}
```



# Output

```
100 + 78 = 178
```

# Funksiya parametrlari

# Types of Functions in C++

returntype function (argument)

Function with no argument  
and no return value

`void function();`

Function with arguments  
but no return value

`void function ( int );`

Function with no arguments  
but returns a value

`int function();`

Function with arguments  
and return value

`int function ( int );`

# Example-1

```
// function declaration  
void greet() {  
    cout << "Hello World";  
}
```

```
#include<iostream>
```

```
void greet() {  
    // code
```

```
}
```

```
int main() {  
    ... ..  
    greet();  
    ... ..  
}
```

function  
call



The diagram illustrates the execution flow of a function call. A blue arrow originates from the `greet();` line within the `main()` function and points to the opening curly brace of the `greet()` function definition. A second blue arrow originates from the closing curly brace of the `greet()` function and points back to the line immediately following the `greet();` call in the `main()` function, indicating the return path.

## Example-2

```
// display a number  
void displayNum(int n1, float n2) {  
    cout << "The int number is " << n1;  
    cout << "The double number is " << n2;  
}
```

```
#include<iostream>
```

```
void displayNum(int n1, double n2) {
```

```
// code
```

```
}
```

```
int main() {
```

```
... ..
```

```
displayNum(num1, num2);
```

```
... ..
```

```
}
```

**function  
call**



The diagram illustrates the execution flow of the code. A blue arrow originates from the closing curly brace of the `main` function and points to the opening curly brace of the `displayNum` function, indicating a return path. Another blue arrow starts from the `displayNum(num1, num2);` line in `main` and points to the opening curly brace of the `displayNum` function, representing the function call. A third blue arrow starts from the opening curly brace of `displayNum` and points back to the `displayNum(num1, num2);` line in `main`, completing the call cycle.

```
#include <iostream>
using namespace std;

// display a number
void displayNum(int n1, float n2) {
    cout << "The int number is " << n1;
    cout << "The double number is " << n2;
}

int main() {

    int num1 = 5;
    double num2 = 5.5;

    // calling the function
    displayNum(num1, num2);

    return 0;
}
```



# Output

```
The int number is 5  
The double number is 5.5
```

# Default qiymatli parametrlar

Default qiymatli parametrlar funksiyaga murojaat qilishda aynan shu parametrlarga qiymat bermasdan funksiyani chaqirish imkonini beradi.

## Example-1

```
#include <iostream>
using namespace std;

void info(string = "Yusuf", int = 20);

int main()
{
    info();
}

void info(string name, int age)
{
    cout << name << " is " << age << " years old";
}
```

## Output

```
Yusuf is 20 years old
```

## Example-2

```
#include <iostream>
using namespace std;

void info(string = "Yusuf",int = 20);

int main()
{
    info("Abror");
}

void info(string name,int age)
{
    cout << name <<" is " << age <<" years old";
}
```

## Output

```
Abror is 20 years old
```

## Example-3

```
#include <iostream>
using namespace std;

void info(string = "Yusuf", int = 20);

int main()
{
    info("Alibek", 25);
}

void info(string name, int age)
{
    cout << name << " is " << age << " years old";
}
```



## Output

```
Alibek is 25 years old
```

# Amaliy mashqlar

Berilgan 3 ta sonning o'rta arifmetigini hisoblab,  
natija sifatida qaytaruvchi funksiya tuzing.

Berilgan 3 ta sonning eng kattasini aniqlab, natija sifatida qaytaruvchi funksiya tuzing.

Berilgan 5 ta sondan manfiylarining yig'indisini aniqlab, natija sifatida qaytaruvchi funksiya tuzing.

Berilgan natural sonning raqamlari yig'indisi hisoblab,  
natija sifatida qaytaruvchi funksiya tuzing.

Foydalanuvchi tomonidan kiritilgan ismga salom beruvchi  
helloName() funksiyasini tuzing.

*Masalan:*

```
helloName("Bob") -> "Hello Bob";
```

```
helloName("X") -> "Hello X";
```

Berilgan 2 ta natural sonning EKUB (eng katta umumiy bo'luvchi) ni hisoblab, natija sifatida qaytaruvchi funksiya tuzing.



Berilgan 2 ta natural sonning EKUK (eng kichik umumiy karrali) ni hisoblab, natija sifatida qaytaruvchi funksiya tuzing.

**E`tiboringiz uchun  
rahmat!**