Function Overloading

Function overloading is a feature of object-oriented programming where two or more functions can have the same name but different parameters. When a function name is overloaded with different jobs it is called Function Overloading. In Function Overloading “Function” name should be the same and the arguments should be different.

The parameters should follow any one or more than one of the following conditions for Function overloading:

- Parameters should have a different type

> add(int a, int b)
>
> *add(double a, double b)*

Below is the implementation of the above discussion:

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


void add(int a, int b)
{
cout << "sum = " << (a + b);
}

void add(double a, double b)
{
    cout << endl << "sum = " << (a + b);
}

// Driver code
int main()
{
    add(10, 2);
    add(5.3, 6.2);
    return 0;
}
```

Output

```
sum = 12
sum = 11.5
```

- Parameters should have a different number

> add(int a, int b)
>
> *add(int a, int b, int c)*

Below is the implementation of the above discussion:

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

void add(int a, double b)
{
    cout<<"sum = "<<(a+b);
}

void add(double a, int b)
{
    cout<<endl<<"sum = "<<(a+b);
}

// Driver code
int main()
{
    add(10,2.5);
    add(5.5,6);

    return 0;
}
```

Output

```
sum = 12.5
sum = 11.5
```

Following is a simple C++ example to demonstrate function overloading.

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

void print(int i) {
        cout << " Here is int " << i << endl;
}
void print(double f) {
        cout << " Here is float " << f << endl;
}
void print(char const *c) {
        cout << " Here is char* " << c << endl;
}

int main()
{
        print(10);
        print(10.10);
        print("ten");
        return 0;
}
```

Output

```
 Here is int 10
Here is float 10.1
Here is char* ten
```

How does Function Overloading work?

- *Exact match*:- (Function name and Parameter)
- *If* a *not exact match is found:–*

->Char, Unsigned char, and short are promoted to an int.

->Float is promoted to double

- *If no match is found*:

->C++ tries to find a match through the standard conversion.

- *ELSE ERROR*