DOUBLE_D

PROGRAMMING HANDBOOK

C++ code

```
#include<iostream>
using namespace std;
int main()
{
  //code goes here
}
```

Output

```
cout << “Hello world”;
```

Input

```
cin >> a;
```

Comments

```
// single line comment
/* multi line comment */
```

Variables

```
int a = 15;
double b = 15.0;
float f = 0.3;
char c = ‘D’;
bool d = true;
string e = “Hi”;
```

Variable scope

Global variable

```
#include<iostream>
using namespace std;
string a = "Hello world";
int main()
{
  cout << a;
}
```

```
#include<iostream>
using namespace std;
string a = "Hello world";
int main()
{
  string a = "Another Hello world";
  cout << a;
}
       // outputs “Another Hello world”
```

Local variable

```
#include<iostream>
using namespace std;
int main()
{
  string a = "Hello world";
  cout << a;
}
```

Constatnt

```
#include<iostream>
using namespace std;
int main()
{
  #define a 100;
  cout << a;
}
```

or

```
#include<iostream>
using namespace std;
int main()
{
  const int a = 100;
  cout << a;
}
```

Data type

Integer, Long, Float, Double, Boolean, Char, String

Modifier types

signed, unsigned, long, short

Qualifier

const, volatile, restrict

Function/Method

```
#include<iostream>
using namespace std;
void my_fun()
{
  cout<<"Hello world";
}
int main()
{
  my_fun();
}
```

Class

Single class

```
#include<iostream>
using namespace std;
class My_class
{
  public:
  my_fun()
   {
      cout<<"Hello world";
   }
};
int main()
{
  My_class obj;
  obj.my_fun();
}
```

Multi class

```
#include<iostream>
using namespace std;
class My_class1
{
  public:
  my_fun1()
    {
       cout<<"Hello world";
    }
};
class My_class2
{
  public:
  my_fun2()
    {
       cout<<"Again Hello world";
    }
};

int main()
{
  My_class1 obj1;
  obj1.my_fun1();
  My_class2 obj2;
  obj2.my_fun2();
}
```

Object

```
int main()
{
  My_class obj;
  obj.my_fun();
}
```

For loop

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


DILANGA D AMARASINGHE
PROGRAMMER
COMPUTER SCIENCE
EASTERN UNIVERSITY - SRI LANKA

E-Mail : doubledamarasinghe@gmail.com
GITHUB : https://github.com/DoubleDAmarasinghe
WhatsApp : +960 96 95 658
LinkedIn : @ Dilanga D Amarasinghe

DOUBLE_D

PROGRAMMING HANDBOOK

```
int main()
{
  for(int i=0; i<10; i++)
    {
       cout<<"Hello world"<<endl;
    }
}
```

While loop

```
#include<iostream>
using namespace std;
int main()
{
  int i = 0;
  while(i<10)
    {
       cout << "Hello world"<<endl;
       i++;
    }
}
```

Do while loop

```
#include<iostream>
using namespace std;
int main()
{
  int i = 0;
  do
    {
      cout<<"Hello world"<<endl;
      i++;
    }
  while(i<10);
}
```

Infinite loop

```
#include<iostream>
using namespace std;
int main()
{
  for(;;)
    {
       cout<<"Hello world"<<endl;
    }
}
```

if, else, elseif statements

```
#include<iostream>
using namespace std;
int main()
{
  int a = 10;
  if(a<8)
   {
    cout<<"Hello world";
   }
  else if(a == 10)
   {
    cout<<"Again hello world";
   }
  else
   {
    cout<<"error";
   }
}
```

Switch

```
#include<iostream>
using namespace std;
int main()
{
  int a = 10;
  switch(a)
   {
         case 5:
            cout<<"Five";
            break;
         case 10:
            cout<<"Ten";
            break;
         case 15:
            cout<<"Fifteen";
            break;
         default:
            cout<<"error";
            break;
    }
}
```

Try, throw, catch

```
#include<iostream>
using namespace std;
int main()
{
  try
   {
     int a = 10;
     if(a>20)
          {
             cout<<"Access granted";
          }
     else
          {
             throw(a);
          }
   }
  catch(int a)
   {
     cout<<"Access denied"<<a;
   }
}
```

Inheritance

Single-level inheritance

```
#include<iostream>
using namespace std;
class animal
{
  public:
  my_fun1()
   {
          cout<<"This is super class"<<endl;
   }
};
class dog: public animal
{
  public:
  my_fun2()
   {
```


DILANGA D AMARASINGHE
PROGRAMMER
COMPUTER SCIENCE
EASTERN UNIVERSITY - SRI LANKA

E-Mail : doubledamarasinghe@gmail.com
GITHUB : https://github.com/DoubleDAmarasinghe
WhatsApp : +960 96 95 658
LinkedIn : @ Dilanga D Amarasinghe

```
        cout<<"This is sub class"<<endl;
   }
};
int main()
{
  dog obj;
  obj.my_fun1();
  obj.my_fun2();
}
```

Multi-level inheritance

```
#include<iostream>
using namespace std;
class animal
{
  public:
  my_fun1()
   {
        cout<<"This is super class"<<endl;
   }
};
class dog: public animal
{
  public:
  my_fun2()
   {
        cout<<"This is sub class"<<endl;
   }
};
class puppy: public dog
{
  public:
  my_fun3()
   {
        cout<<"This is sub-sub class"<<endl;
   }
};
int main()
{
  puppy obj;
  obj.my_fun1();
  obj.my_fun2();
  obj.my_fun3();
}
```

Multiple inheritance

```
#include<iostream>
using namespace std;
class animal
{
  protected:
  public:
  void my_fun1()
   {
        cout<<"This is super class"<<endl;
   }
};
class dog
{
  protected:
  public:
  void my_fun2()
   {
        cout<<"This is sub class"<<endl;
   }
};

class puppy: public animal, public dog
{
  public:
  void my_fun3()
   {
        cout<<"This is sub-sub class"<<endl;
   }
};

int main()
{
  puppy obj;
  obj.my_fun1();
  obj.my_fun2();
  obj.my_fun3();
}
```

DILANGA D AMARASINGHE
PROGRAMMER
COMPUTER SCIENCE
EASTERN UNIVERSITY - SRI LANKA

E-Mail : doubledamarasinghe@gmail.com
GITHUB : https://github.com/DoubleDAmarasinghe
WhatsApp : +960 96 95 658
LinkedIn : @ Dilanga D Amarasinghe