

BASICS OF C++

- Include Headers

> # include < headerfile >

- Common Headers

> iostream , istream , math , cctype , string

- Name Space

> Using namespace std.

- Datatype

> int , char , float , double , void , bool.

- Comments

> // comment text

/* Multi-line comment text */

ATUL KUMAR (LINKEDIN).
NOTES - GALLERY (TELEGRAM).

- Relational operators

> < = (less than or equal to), < (less than), > (greater than),
> = (greater than or equal to), == (equal to equal to).

- Logical operators

> || (logical OR), && (logical AND), ! (logical NOT)

- Pointers

> int * ptr

// Define Pointer

ptr = & var

// ptr set to address of var

var 2 = * ptr

// set var 2 , to value of var 1.

- For loop

- `for (< initialize > ; < condition > , < update >)
{ < statement > ; }`

- While loop

- `while (< condition >)
{ < statement > ; }`

- If else

- `if (< condition >)
{ < statement 1 > ; }
else
{ < statement 2 > ; }`

ATUL KUMAR (LINKEDIN).
NOTES - GALLERY (TELEGRAM).

- Do - While loop

- `do { < statement > ; }
while (< condition >);`

- Arrays

- `// New 5 element Array
int myArray [5];
// Array index starts at 0
// Access 3rd element
myArray [2] = var ;`

- I/O operators

- `>> // Input Operator
<< // output Operator
cin >> var 1, var 2, var 3 ;
cout << "Text : " << var 1 << var 2 << end + ;
cin.get (char * buffer, streamsize num char);`

• Switch Statement

```
> Switch (< expression > )  
{  
    case < constant 1 >;  
        < statement sequence 1 >;  
        break;  
  
    case < constant 2 >;  
        < statement sequence 2 >;  
        break;  
  
    case < constant n+1 >;  
        < statement sequence n+1 >;  
        break;  
  
    default :  
        < statement sequence n >;  
        break;  
}
```

ATUL KUMAR (LINKEDIN).
NOTES - GALLERY (TELEGRAM).

• File I/O

> fstream File

```
file.open ("filename" << file mode constant > );
```

// Reads and writes like cin and cout.

```
file >> var;
```

```
file << "Text" << var << endl;
```

// Read Entire Line

```
getline (file, line);
```

// Reading writing binary data.

```
file.read (memory-block, size);
```

```
file.write (memory-block, size);
```

```
file.close ();
```


- File Mode Constants

- > ios::in // opens file for reading
- ios::out // opens file for writing
- ios::ate // seeks EOF, I/O operations can occur.
- ios::app // causes output to be appendent at EOF.
- ios::trunc // Destroys previous contents.
- ios::nocreate // causes open() to fail if doesn't exists
- ios::noreply // causes open() to fail if already exist.

- Function Prototype

- > < return_data_type > < function_name > (Parameter list)
{ body of function }

- Class Prototype

- > class < class_name >
{
 public:
 // method-prototypes
 protected:
 // method-prototypes
 private:
 // method-prototypes
 // data-attributes
}

ATUL KUMAR (LINKEDIN).
NOTES-GALLERY (TELEGRAM).

- Structure Prototype

- > struct < structure_name > {
 member_type 1 member_name 1;
 member_type 2 member_name 2;
}
 < object_name > ;

- Accessing Data Structure

- > // Access member variable from struct/class.

mystruct.member var 1 = var ;

// Call class method

myclass.method 1 (args);

// Pointer to struct / class

mystructType * Ptr ;

Ptr = &.mystruct ;

Ptr -> membervar 1 = var ;

ATUL KUMAR (LINKEDIN).
NOTES-GALLERY (TELEGRAM).