OOPs (file handling)

C++ : File I/O [Reading & writing to a file]

3 useful classes

1) fstreambase } → all comes in #include<fstream>
2) ifstream } → derived from ① fstreambase
3) ofstream }

Read operation / write operation

* Read operation

```
ifstream in("this.txt");
string str;
in>>str; // Just like cin
```

→ str mein aa jaega data.

→ but this will only take a single word agar fir se poora sentence chahiye toh

```
getline(in, str);
```

* write operation

```
ofstream out("this.txt");
string st = "Harry";
out<<st; // write to a file this.txt.
```

example prog:

```
int main(){
    Ofstream out;
    out.open("sample.txt");
    out<<"This is me\n";
    out<<"This is also me";
    out<<"This is also me";
    out.close();

    ifstream in;
    string st;
    in.open("sample.txt");
    while(in.eof()==0){
        getline(in,st);
        cout<<st<<endl;
    }
    in.close();
    return 0;
}
```

→ append karke jaega file mein.

Jab file ke end mein jaega

objn.eof()==1

at last.

file Modes

There two methods (constructor, open()) take one argument only, But we can pass Two arguments.

second argument is file mode parameter.

eg

```
stream-object.open("filename", mode);
```

- ios::app (append to end of File)
- ios::ate (Go to End of file on opening)
- ios::binary (Binary file)
- ios::in (open the file for reading only)
- ios::nocreate (open fails if file doesn't exist)
- ios::noreplace (open fails if file already exists)
- ios::out (open file for writing only, automatically open in trunc)
- ios::trunc (Delete the contents of the file if exists)

If we do not specify the second argument then by default the argument passed is

ios::in for ifstream (means read only)

ios::out for ofstream (means write only)

Difference between C++ and Java.

1) C++ is only a compiled language while Java is both compiled and interpreted.

2) C++ ~~compiler converts~~ is platform dependent whereas Java is platform independent.

3) Java supports inheritance except for multiple inheritance whereas C++ supports all inheritance including multiple inheritance

4) Java doesn't support destructor where as C++ does.

5) Java doesn't support pointers, operator overloading, structure.