

[16]io

서버에 올라간 프로그램을 특정 폴더에 카피하는 걸 꼭 만들어야 "파일첨부" 게시판 작성 가능
그냥 콘솔이 아니라 특정한 아이(파일)을 꼭집어 가지고와서 입출력 하기 위해 stream이 필요함
Stream은 빨대(꼭고 제거하고) 의 의미

스트림을 연다 = 파일을 연다

Input stream or reader 통해 파일읽기

Input stream(1바이트) reader(2바이트) 영어는 1바이트 우리언어는 그게 아님

하지만 ostream을 더 많이 쓰긴 함

```
public static void main(String[] args) {
    InputStream is = null; // null값을 주는 이유 : null을 주지 않으면 finally try 에서 is.close()에 만약, is에 아무값도 안들어
    //위예다가 변수선언 한 이유는 finally에서도 써주기 위해..
    try {
        is = new FileInputStream("txtFile/inTest.txt"); //----- 1.파일을 연다
        while(true) { //파일의 언제까지 읽을줄 모르니, while를 써서 파일의 끝까지 읽어주기
            int i = is.read(); //여기서 나오는 예외 처리 해줘야됨. try절 안에 try절 추가 하면 안되고, 밑에 캐치절로 추가해주자 //--2.
            //(1byte씩 읽을거니 int를 써줘야됨)
            if(i== -1)break; //파일의 끝이면 반복문break //파일의 끝이라는 코드값을 읽으면 i가 -1를 리턴함.. 따라서 -1즉 파일의 끝을 만나
            System.out.print(i); // i의 아스키코드가 나옴.. 싶으면 형변환 해줘야됨
            System.out.print((char)i + "(" + i + ")"); //형변환 안하면 아스키코드 나오는 것
        }
        System.out.println("\n끝");
        //여기서 파일을 닫아버리면 2번 예러가 났을때 닫을수가 없음. 2번 수행하다말고 catch절로 떨어져버리면 안됨
    } catch (FileNotFoundException e) {
        System.out.println("파일을 못 찾은 경우 예외 : " + e.getMessage());
    } catch (IOException e) {
        System.out.println("파일을 읽을 수 없는 경우 예외 : " + e.getMessage());
    } finally {
        //3. 파일을 닫는다 //예외가 발생해도 finally 실행, 예외가 발생하지 않아도 finally 실행
        //finally에 is 변수가 없으면 안되니, InputStream is는 맨 위로 빼주자
        try {
            if(is!=null) is.close(); //is가 null이면 close 실행하지 않고 끝냄
        } catch (IOException e) {
            System.out.println(e.getMessage());
        }
    }
}
```

파일을 열고, 파일을 읽고, 꼭 닫아줘야함

Inputstream은 추상 class라 객체 생성 불가능

하위 class중 fileinputstream으로 객체 생성해야함

예외가 발생하니 (없는 파일 불러올수도 있으니) try catch는 항상 해주기

```

public class Ex03_outputStream {
    public static void main(String[] args) {
        OutputStream os = null;
        try {////////////////////////////////////↓true를 안쓰면 덮어씀. true를 쓰면 append되서 ㄹ
            os = new FileOutputStream("txtfile/outtest.txt", true); //1.파일을 연다
            byte[] bs = { 'H', 'e', 'l', 'l', 'o' };
            for(int i = 0 ; i<bs.length ; i++) { //애는 확장포문 넣어도 됨 2.write 한다.
                os.write(bs[i]); // 애는 콘솔 아니고 파일에 추가 되는 거임
            } //for문 안쓰고 os.write(bs) 로 바로 써도 가능
            System.out.println("파일 출력 성공 ");
        } catch (FileNotFoundException e) {
            System.out.println(e.getMessage());
        } catch (IOException e) {
            System.out.println(e.getMessage());
        } finally {
            try {
                if(os !=null)
                    os.close();
            } catch (IOException e) {
                System.out.println(e.getMessage());
            }
        }
    }
}

```

output도 똑같이 파일을 열고, 파일을 읽고 파일을 닫아야함