

12 라이브러리 활용하기

입출력작업

1 개요

- ▶ 입출력 원리
 - 자바는 외부 데이터와 입출력 작업을 할 때 스트림을 사용함
 - 입력 스트림 : 외부 데이터를 읽어오는 기능
 - 출력 스트림 : 외부로 데이터를 출력하는 기능
- ▶ 입출력 API
 - 자바 API는 java.io 패키지에 외부 데이터를 입출력할 수 있는 API들을 제공함
 - ~InputStream 또는 ~Reader 이름의 객체는 입력 스트림 객체
 - ~OutputStream 또는 ~Writer 이름의 객체는 출력 스트림 객체

2 기본 입출력 작업

- ▶ 파일 입출력
 - FileInputStream : 파일로부터 데이터를 입력 받기 위한 입력 스트림
 - FileOutputStream : 파일로 데이터를 출력하기 위한 출력 스트림
- ▶ 표준 입출력
 - System.in : 표준 입력 스트림
 - System.out : 표준 출력 스트림
- ▶ 필터 스트림 활용
 - 입출력 작업 시 추가적인 기능을 사용하기 위한 객체
 - BufferedReader는 줄단위로 읽기가 가능한 readLine() 메소드를 제공함

3 다양한 입출력 방식

- ▶ 바이트 배열
 - 입력 스트림의 read(바이트 배열)를 사용하여 바이트 배열의 데이터를 읽을 수 있음
 - ByteArrayOutputStream은 바이트 배열로 데이터를 출력하는 출력 스트림
- ▶ File
 - 파일 또는 디렉터리에 관련된 기능을 지원하는 객체
- ▶ RandomAccessFile
 - 파일의 임의의 위치에 데이터를 읽고 쓰기 위한 객체

12. 라이브러리 활용하기

입출력작업

- › 프로퍼티 파일
 - 확장자가 property인 파일은 텍스트 파일로서 데이터가 name=value 형태로 저장됨
 - Properties는 프로퍼티 파일에 데이터를 읽고 쓰기 위한 객체
- › FileChannel
 - 고속으로 파일의 데이터를 읽고 쓸 수 있도록 지원하는 객체
- › 압축 파일
 - ZipOutputStream : 파일들을 압축하여 압축 파일을 생성하는 객체
 - ZipInputStream : 압축된 파일들을 해제하는 객체

4 객체 직렬화

- › Serializable
 - 인터페이스를 상속받은 자바 객체는 입출력 작업을 할 수 있음
- › serialVersionUID 필드
 - 객체를 직렬화하여 입출력 작업을 할 때는 객체의 고유한 인식번호를 부여할 수 있음
 - 객체의 고유한 인식번호를 저장하는 필드임