**로고학습일지**

**kt ds University 자바 기반의 데이터 사이언티스트 양성과정**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 학습일시 | 2018. 07. 18 (수) | 장소 | kt ds University B관 201호 | **시 간** | 09:00~18:00 |
| 학습범위 | UI/UX | | | | |
| 작 성 자 | 류복기 | | | **강 사** | 장민창 강사 |

|  |  |
| --- | --- |
| 학습안건 | - 주석  - 형 변환  - data structure; Array, List |

|  |  |
| --- | --- |
| 학습내용 | 내용 |
| 1. 주석   1)매우 복잡한 코드에 설명이 필요할 때 사용  2)3가지형태로 나뉨  - // -> 위, 아래, 오른쪽에 한 줄로 간단하게 사용  Single line  - /\* \*/ -> 코드 위에 여러 줄 작성시  Multi line  -/\*\* \*/ -> 사용가이드를 작성할 때 사용   1. String   1)여러 개의 문자를 저장하려면 배열을 이용해야 함.  2)일일이 모든 문자열을 배열을 통해 저장할 수 없으므로 JAVA는 String 이라는 문자열을 저장할 수 있는 특수한 타입을 제공.   1. Type Casting(형 변환)   1)java는 사칙연산의 결과를 자동으로 변환해 저장할 수 있도록 합니다.  2)형 변환이란, 현재 할당되어야 할 값이 변수의 타입과 맞지 않을 경우 해당 변수에 값이 들어갈 수 있게 data type을 바꿔주는 것  - 묵시적(자동) 형 변환  **작은 변수**에서 **큰 변수**로 갈 때,  Byte -> short -> int -> long -> float -> double -> String  Boolean -> String  Char -> String  - 명시적(수동) 형 변환  **큰 변수**에서 **작은 변수**로 갈 때  계산만을 위해 잠시 동안 형을 바꿔주고 싶을 때   1. Array   공통적인 성격의 변수가 필요할 때 변수를 선언해 여러 개의 값을 저장하고 싶을 때 사용.  예를 들어, 4과목의 성적을 한 사람의 성적에 담고 싶을 때  배열의 선언 형식  예시) 자료형[] 배열이름 = new 자료형[저장할 수 있는 변수의 개수];   * 여기서 저장할 수 있는 변수의 개수가 배열의 크기가 됨.   배열에 data 저장  배열이름[index] = 데이터 값   * 이때 데이터의 형식은 배열의 형식과 맞아야 한다. * 한번 정의된 배열의 크기는 줄이거나 늘리거나 할 수 없음.   예를 들어, 배열의 크기를 한번 정의하고 다시 재정의 하는 경우 오류 발생  배열의 길이  Arrayname.length -> 배열의 길이값 리턴해줌.  배열의 데이터 값 추출  Arrayname[index] -> index에 해당하는 데이터 값 추출   1. List   배열의 크기를 원하는 대로 추가/삭제 할 수 있도록 지원함.  리스트의 생성   * 클래스 위에 import.java.util.List; import.java.util.ArrayList; 작성 필요! * 리스트 중에서는 ArrayList를 많이 사용   List<wrapper class> 리스트의 이름 = new ArrayList<wrapper class>();  Wrapper class란 <>(제네릭)에 primitive type을 입력할 수 없으므로 원시타입을 클래스화 시킨 것   * 원시타입을 감싸는 클래스라고 생각하자!   -primitive type이란?  Byte, short, int, long, float, double, char를 의미!  -wrapper class의 종류  Byte, Short, Integer, Long, Float, Double, Char  리스트에 데이터 삽입  리스트이름.add(입력할 데이터) -> 입력할 데이터는 리스트의 자료형과 맞아야 함.  리스트 데이터 제거  리스트이름.remove(index) -> 제거할 데이터의 인덱스를 입력!  첫 번째 인덱스만 계속 제거하거나 마지막 인덱스만 제거해야함.  예시) List<Integer> numberList = new List<Integer>();  numberList.add(1);  numberList.add(2);  numberList.add(3);  일때, numberList.remove(0); numberList.remove(1); numberList.remove(2);를 입력하면 오류발생  numberList.remove(0);을 실행시 첫번째 데이터인 1이 지워짐. 이후 리스트의 사이즈가 2개로 줄어듦. 이후 numberList.remove(1);실행시 두번째 데이터인 3이 지워짐. 이후 리스트의 사이즈가 1개로 줄어듦. 이후 numberList.remove(2);실행시 오류발생!! -> 두번째 인덱스가 존재하지 않기 때문!  리스트 데이터 값 추출  리스트명.get(index);  리스트의 길이 값 추출  리스트명.size();   1. Map   맵이란?  파이썬의 Dictionary와 같이 key값과 value값으로 구성되있음.  맵의 생성   * Class 위에 import.java.util.Map; import.java.util.HashMap; 작성 필요 * Map중에서는 HashMap을 많이 사용   Map<key의 wrapper class, value의 wrapper class> mapname = new HashMap<key의 wrapper class, value의 wrapper class>();  맵의 데이터 삽입  Mapname.put(key값, values값); -> 여기서 key값과 value값은 wrapper class의 자료형과 맞아야 함.  맵의 데이터 제거  Mapname.remove(key값); -> key를 List의 인덱스처럼 사용  맵의 데이터 추출  Mapname.get(key값); -> key값을 통해 value추출   * 실습예제  1. CelToFah project   화씨 <-> 섭씨 변환기   * String의 비교를 사용하기 위해서는 == 대신 string의 이름.equals(“비교하고자 하는 이름”)을 사용한다.  1. Average\_Calculator Project   수학, 물리, 영어, 프로그래밍 4가지 과목의 평균을 구하기   1. Calculate average score about each person 2. 인원별 4개 과목 점수를 Map에 저장 3. 인원별 과목 점수의 평균을 출력  * 어떤 것을 key값으로 어떤 것을 value값으로 할 지 정해야 함. * 여기서는 사람명이 key값이 되고 4개 과목 점수가 value값이 됨. * 그래서 key의 Wrapper class는 String을 사용하고 value의 wrapper class는 List를 사용한다. * 첫번째, 맵을 생성 * 두번째, 리스트를 생성하여 과목별 점수를 리스트에 담는다 * 세번째, 맵에 인원별 점수를 리스트로 해서 추가 * 네번째, 맵에서 추출한 인원별 점수를 새로운 리스트 변수로 담는다. * 다섯번째, 인원별 리스트의 데이터를 추출해, 합계와 평균을 구한다. |

|  |  |
| --- | --- |
| 학습외의 내용 |  |