1단계 : 종료

2단계 : 등록

3단계 : 등록 시 생성된 명언번호 노출

4단계 : 등록할 때 마다 생성되는 명언번호 증가

5단계 : 목록 만들기

6단계 : 해당 번호 명언 삭제

7단계 : 존재하지 않는 명언 삭제에 대한 예외처리

출처 : [글 13123 | S-CODE](https://www.s-code.io/usr/p/13123)

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명  
텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

package Testing;  
  
import java.util.ArrayList;  
import java.util.List;  
import java.util.Scanner;  
  
  
public class test {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.*in*);  
 List<wiseSaying> list = new ArrayList<wiseSaying>();  
 int num = 0; // id 넣기 위한 숫자.  
  
 System.*out*.print("== 명언 앱 ==" + '\n' );  
 while(true){  
  
 System.*out*.print("명령) ");  
 String button = sc.nextLine();  
  
 if (button.equals("종료")) break; // 종료 버튼  
 else if (button.equals("등록")) {  
 System.*out*.print("명언 : ");  
 String content = sc.nextLine();  
 System.*out*.print("작가 : ");  
 String author = sc.nextLine();  
 System.*out*.println();  
 num++;  
 System.*out*.println(num + "번 명언이 등록되었습니다");  
 list.add(new wiseSaying(num ,content,author)); // arraylist에 객체 삽입을 위해 해당 명령어 생성 (또한 class 객체에 생성자 해줘야함)  
 } // 정보 등록  
  
 else if (button.equals("목록")) {  
 for (int i = 0; i < list.size(); i++) {  
 System.*out*.println(list.get(i).returnNum() + " " +  
 list.get(i).returnContent() + " " + list.get(i).returnAuthor()); // 리스트 확인을 위해 get() 명령어 사용  
 }  
  
 } // 3. 목록 확인  
 else if(button.equals("삭제")) {  
 System.*out*.print("?id=");  
 String delete = sc.nextLine();  
 int deleteNum = Integer.*parseInt*(delete);  
 boolean deleteEnter = false;  
  
  
 for (int i = 0; i < list.size(); i++) {  
 System.*out*.println("확인용 : " + list.get(i).num);  
 System.*out*.println("실제 위치 : " + i);  
 if (list.get(i).num == deleteNum) { // deleteNum에 입력한 숫자는 remove할 숫자랑 자리가 안맞으니 -1 통해 조절해준다.  
 System.*out*.println("저 지금 여기 들어왔어요 ~~~ : " + list.get(i).num);  
 deleteEnter = true;  
 System.*out*.println("삭제 확인용 : " + list.get(i).returnNum());  
 System.*out*.println("삭제 확인용 : " + list.get(i).returnContent());  
 System.*out*.println("삭제 확인용 : " + list.get(i).returnAuthor());  
 list.remove(list.get(i)); // 다시는 remove시 get.num 하지말자... list.get 이 자리가 아닌 .num의 자리를 인식한다..  
 // 예를 들어 두 번째를 삭제하고 싶다. 형식상 2라는 곳을 삭제한다 입력하니 실제 list에 저장된 곳은 1번쨰다.  
 // 나는 여기서 deleteNum - 1 해줌으로써 list실제 자리에 맞춰줬고 그 부분을 삭제하려 한다.  
 // 하지만 remove(list.get(i).num) 을 해주면 두 번째 자리 num은 3이 존재하니 그 숫자를 인식하게 된다.  
 break;  
 }  
// if (i >= deleteNum) {  
// list.get(i).num--;  
// System.out.println(list.get(i).num);  
// }  
 }  
  
 if (deleteEnter) {  
 for (int i = 0; i < list.size(); i++) {  
 if (i >= (deleteNum - 1)) {  
 list.get(i).num -= 1;  
 System.*out*.println(list.get(i).num);  
 } // 데이터 하나 삭제 시, 삭제한 숫자 위의 숫자들을 앞당겨와야 하기 떄문에 앞 숫자를 -1을 한다.  
 }  
 num--;} // 삭제 처리 정상 진행 될 시 true  
 else System.*out*.println(deleteNum + "번 명언은 존재하지 않습니다."); // 그게 아니면 false  
 } // 4. 해당 인덱스 삭제  
 else if (button.equals("수정")) {  
 System.*out*.print("수정? id=");  
 String paste = sc.nextLine();  
 int pasteNum = Integer.*parseInt*(paste);  
  
 for (int i = 0; i < list.size(); i++) {  
 if (list.get(i).num == pasteNum) {  
 System.*out*.println("명언 기존 : " + list.get(i).returnContent());  
 System.*out*.print("명언 : ");  
 String content = sc.nextLine(); list.get(i).content = content; // 명언 수정  
 System.*out*.println("작가(기존) : " + list.get(i).author);  
 System.*out*.print("작가 : ");  
 String author = sc.nextLine(); list.get(i).author = author;  
 }  
 } // 5. 해당 id 내용 수정  
 } else System.*out*.println("잘못 된 명령어");  
 }  
 }  
}  
  
class wiseSaying {  
 int num = 0;  
 String content;  
 String author;  
  
 public wiseSaying(int num, String content, String author) {  
 this.num = num; this.content = content; this.author = author;  
 } // arraylist에 메인에서 선언된 num, content, author을 집어넣기 위해 생성자 생성  
 public int returnNum() {  
 return num;  
 }  
 public String returnContent() {  
 return content;  
 }  
 public String returnAuthor() {  
 return author;  
 }  
}