

실습) 주소록 프로그램 1

클래스 구현 (Person, Student, Worker)

◆ Person 클래스

- 이름, 전화번호 인스턴스 변수
- 이름, 전화번호 반환 메소드

◆ Student 클래스 (Person 클래스로부터 상속)

- 학과, 학번 인스턴스 변수
- 학과, 학번 반환 메소드

◆ Worker 클래스 (Person 클래스로부터 상속)

- 부서, 사번 인스턴스 변수
- 부서, 사번 반환 메소드

◆ Student 또는 Worker 인스턴스 10명 생성

◆ 구분(타입)은 알아서...

메인코드 구현 (주소록 생성 및 출력)

◆ 이름 10개, 전공 및 부서 리스트 코드에 고정

- 예시)

```
nameList = ["홍동현", "서상욱", "도찬호", "이대희", "최기태", "김동호", "김준식", "이영석", "김승주", "김종범"]
majorList = ["컴퓨터학부", "경영학부", "통계학과", "기계공학부"]
departList = ["개발연구팀", "전략기획팀", "인사관리팀"]
```

◆ 학생과 회사를 랜덤하게 생성하여 주소록 리스트 생성

- 랜덤하게 생성한 타입에 따라 majorList, departList에서 랜덤하게 생성 (이름은 순차적으로 가져와도 됨)
- 전화번호 랜덤하게 생성 010-XXXX-XXXX
- 학생일 시 학번 랜덤하게 생성 201XXXXXXXX (10자리)
- 회사원일 시 사번 랜덤하게 생성 2019XXX (7자리)

◆ 생성된 전체 주소록 출력 (10명)

- [학생] 이름: XXX 전화번호: 010-XXXX-XXXX 학과: XXXXX 학번: 201XXXXXXXX
- [사원] 이름: XXX 전화번호: 010-XXXX-XXXX 부서: XXXXX 사번: 2019XXX
-

메인코드 구현 (주소록 검색)

◆ 이름으로 검색 → 전화번호 및 정보 출력

- ‘끝’ 입력 시까지 여러 번 검색 가능

◆ 화면 예시

```
----- 주소록 프로그램 -----
[사원] 이름: 홍동현      전화번호: 010-7181-4937   부서: 개발연구팀   사번: 2019072
[사원] 이름: 서상욱      전화번호: 010-7549-6028   부서: 개발연구팀   사번: 2019072
[사원] 이름: 도찬호      전화번호: 010-8860-8044   부서: 인사관리팀   사번: 2019063
[학생] 이름: 이대희      전화번호: 010-7628-1046   학과: 컴퓨터학부   학번: 2016713443
[사원] 이름: 최기태      전화번호: 010-8956-7281   부서: 전략기획팀   사번: 2019004
[학생] 이름: 김동호      전화번호: 010-9227-6711   학과: 경영학부     학번: 2016248770
[학생] 이름: 김준식      전화번호: 010-9964-8057   학과: 통계학과     학번: 2016482967
[학생] 이름: 이영석      전화번호: 010-9395-7947   학과: 기계공학부   학번: 2015477948
[학생] 이름: 김승주      전화번호: 010-1898-4846   학과: 경영학부     학번: 2017320102
[학생] 이름: 김종범      전화번호: 010-3944-1182   학과: 기계공학부   학번: 2014733284
-----
찾을 사람 이름은 ? 홍동현
전화번호 : 010-7181-4937
부서: 개발연구팀
사번: 2019072
찾을 사람 이름은 ? 이대희
전화번호 : 010-7628-1046
학과: 컴퓨터학부
학번: 2016713443
찾을 사람 이름은 ? 이영석
전화번호 : 010-9395-7947
학과: 기계공학부
학번: 2015477948
찾을 사람 이름은 ? 끝
주소록 프로그램을 종료합니다.
```