

파이썬

상명대학교 융합공과대학 지능·데이터융합학부

휴먼지능정보공학전공

상명대학교 바이오헬스혁신공유대학

dkim@smu.ac.kr

교수소개

- 전공

- 생체공학전공 (공학박사)

- 소속

- 휴먼지능정보공학전공 주임교수
- 지능·데이터융합학부 학부장
- 대학원 지능정보공학과 학과장
- 상명대학교 바이오혁신공유대학
- 서울캠퍼스 지능정보기술연구소 소장 (<https://ai.smu.ac.kr>)

- 강의

- 지능정보공학설계 / 이산수학
- 프로그래밍 (윈도우, 웹, 네트워크, 데이터베이스, 모바일)
- 프로그래밍 디자인 / 컴퓨팅사고력과SW코딩

과목소개

• 과목개요

- 파이썬 프로그래밍의 기초 내용을 학습 (이론과 실습)
- 문제해결이 가능한 실제적인 코딩 기법과 문제해결 과정에 따른 기초적인 알고리즘을 적용

• 과목목표

- 파이썬 프로그래밍 기초 능력을 향상시키고 문제해결이 가능한 실제적인 코딩 기법과 기초알고리즘의 학습이 가능하도록 함
- 연관된 주제에 대한 적절한 문제해결과 SW 코딩이 가능한 전공 역량을 갖추도록 함

과목소개

• 과목개요

- 학부기초 전공선택(3학점): 이론과 실습으로 구성
- 설계(실습): 파이썬 프로그래밍
- 교재: 수업강의자료
- 평가방법
 - 중간 30%, 기말40%, 출석10%, 과제물 20%
- 참고사항
 - 과제 제출기한 지나면 채점 없음
 - 부정행위(시험부정, 보고서/발표 무단복사) 수업시간 1/3 무단결석 F학점
- 선수과목

강의개요

- 강의소개 및 프로그래밍 개념
 - 프로그래밍과 컴퓨팅사고력 소개
 - 프로그래밍 맛보기
- 변수, 자료형, 연산, 함수
 - 코딩과 기초실습
- 조건문, 연산자
 - 코딩과 기초실습
- 반복문
 - 코딩과 기초실습
- 함수, 매개변수
 - 코딩과 기초실습
- 중간고사

강의개요

- 자료형, 리스트
 - 코딩과 기초실습
- 자료형, 튜플
 - 코딩과 기초실습
- 자료형, 딕셔너리
 - 코딩과 기초실습
- 실습예제
 - 코딩과 기초실습
- 파일읽고 쓰기
 - 코딩과 기초실습
- 객체지향 프로그래밍
 - 코딩과 기초실습
- 기말고사

소프트웨어 이해하기

- 소프트웨어
 - 컴퓨터 시스템을 효율적으로 운영하기 위해 개발된 프로그램의 총칭
- 소프트웨어 종류
 - 시스템 소프트웨어
 - 하드웨어를 제어할 수 있는 소프트웨어, 대표적으로 운영체제(OS)
 - 응용소프트웨어
 - 시스템 소프트웨어의 도움을 받아 사용자가 원하는 작업을 처리해주는 소프트웨어, 대표적으로 워드프로세서

프로그래밍 이해하기

- 프로그램
 - 진행 계획이나 순서. 또는, 그 목록. '계획(표)', '차레(표)'.
 - 음악회 프로그램
- 컴퓨터 프로그램
 - 어떤 문제를 해결하기 위해 컴퓨터에게 주어지는 처리 방법과 순서를 기술한 일련의 명령문의 집합

프로그래밍 이해하기

- 프로그래밍

- 프로그래밍(programming)은 프로그램을 작성하는 것
- 프로그램을 작성하는 사람을 프로그래머(programmer)
- 프로그래밍 작성 과정이 프로그래밍 또는 코딩(coding)
- 컴퓨터가 이해할 수 있는 규칙에 따라 프로그램 수행절차를 프로그래밍 언어로 작성하는 것

- 프로그래밍언어

- 컴퓨터 시스템을 동작시키는 소프트웨어를 작성하기 위한 형식언어
- 컴퓨터를 이용하여 특정 문제를 해결하기 위한 프로그램을 작성하기 위해 사용되는 언어
- 컴퓨터 소프트웨어를 만드는 언어

프로그래밍 이해하기

- 프로그래밍언어

- 컴퓨터 시스템을 동작시키는 소프트웨어를 작성하기 위한 형식언어
- 컴퓨터를 이용하여 특정 문제를 해결하기 위한 프로그램을 작성하기 위해 사용되는 언어
- 컴퓨터 소프트웨어를 만드는 언어

- 프로그래밍언어 분류

- 저급언어

- 기계어 (Machine Language) : 컴퓨터가 직접 이해할 수 있는 언어
- 어셈블리어 (Assembly Language) : 기계어와 1:1로 대응되는 기호로 이루어진 언어

- 고급언어

- 실행을 위해서는 번역하는 과정이 필요
- 상이한 기계에서 소스 수정 없이 실행이 가능
- C, C++, Java, Python 등

프로그래밍 이해하기

- 프로그래밍언어 구분

- 컴파일언어

- 컴파일: 고급언어로 작성된 명령문을 기계어로 바꾸는 단계
 - 프로그램 전체를 한번에 기계어로 번역하는 방식
 - 고급 언어로 작성한 원시코드를 기계어인 목적코드로 번역
 - 목적코드는 로더에 의해 메모리에 로드된 후 실행하여 결과값
 - C, C++, Java

- 인터프리트언어

- 프로그램을 한 행씩 읽어 번역과 동시에 실행을 수행하는 방식
 - Python, JavaScript

프로그래밍 이해하기

- 코딩

- ‘컴퓨터 프로그램을 수행하는 절차를 적어 둔 명령어들이 코드(code)를 작성하는 행위’
- 코딩은 ‘문제해결을 위한 절차와 과정을 설정하고, 그것을 실행 가능한 프로그램으로 작성하는 일’
- 코딩의 궁극적 목표는 주어진 문제를 제대로 해결하는 일
- 코딩에 앞서 문제해결을 위한 방법을 먼저 구상
 - 효율적인 코딩은 먼저 알고리즘(algorithm)부터 구상
 - 알고리즘은 ‘어떤 작업을 수행하는데 있어 적합한 절차와 과정’

프로그래밍 이해하기

- 알고리즘

- 어떠한 문제를 해결하기 위한 여러 동작들의 모임
- 프로그래밍 알고리즘: 프로그래밍 언어를 사용하여 어떠한 문제를 해결하기 위한 명령어 모임

- 알고리즘 전제조건

- 입력 : 외부에서 제공되는 자료가 존재
- 출력 : 적어도 2개 이상의 서로 다른 결과
 - 즉 모든 입력에 하나의 같은 출력이 나오면 안 됨
- 명확성 : 수행 과정은 명확하고 모호하지 않은 명령어로 구성
- 유한성 : 유한 번의 명령어를 수행 후 유한 시간 내에 종료
- 효율성 : 모든 과정은 명백하게 실행 가능(검증 가능)

프로그래밍 이해하기

- 컴퓨터 프로그램
 - 어떤 문제를 해결하기 위해 컴퓨터에게 주어지는 처리 방법과 순서를 기술한 일련의 명령문의 집합
- 컴퓨터 프로그래밍
 - 컴퓨터가 이해할 수 있는 규칙에 따라 프로그램 수행절차를 프로그래밍 언어로 작성하는 것
- 프로그래밍 알고리즘
 - 프로그래밍 언어를 사용하여 어떠한 문제를 해결하기 위한 명령어 모임

프로그래밍 기본요소

- 데이터(자료): 프로그램을 운용할 수 있는 형태로 기호화 숫자화 한 자료(데이터)
 - 숫자, 문자
- 변수: 데이터(자료)를 저장하는 공간 (이름, 형태가 있음)
 - 정수형, 실수형, 문자, 문자열(str)형, 리스트(list)형, 불(bool)형, 튜플(tuple), 집합(set)형, 사전(dict)형 등
- 상수: 항상 같은 값을 가지는 수나 문자 데이터(자료) 자체
 - 숫자, 문자

프로그래밍 기본요소

- 자료형: 데이터(자료)를 표현하는 방법(형태가 있음)
 - 정수형, 실수형, 문자, 문자열(str)형, 리스트(list)형, 불(bool)형, 튜플(tuple), 집합(set)형, 사전(dict)형 등
- a=10
 - a(변수, 정수형) = (대입연산자) 10(데이터, 자료, 상수)

프로그래밍 기본요소

- 자료구조: 데이터를 조직(생성), 저장, 표현하는데 요구되는 방식과 이를 구현하는데 필요한 알고리즘에 관한 이론
 - 저장, 탐색, 삭제
- 연산자: 프로그램의 산술식이나 연산식을 표현하고 처리하기 위해 제공되는 기호
 - 대입연산, 산술연산, 증감연산, 관계연산, 논리연산, 비교연산, 삼항연산
- 명령문: 프로그램이 특정 작업을 수행하도록 프로그래밍 언어로 작성하는 명령
 - 선언문, 대입문, 함수(호출)문, 반복문(for, while), 조건문(if~else, if~elseif, switch case), 분기문(break, continue)

프로그래밍 기본요소

- 함수: 하나의 특별한 목적의 작업을 수행하기 위해 독립적으로 설계된 코드 모음
 - 내장함수(매개변수, 반환 값), 사용자정의함수(매개변수, 반환 값), 라이브러리(매개변수, 반환 값)
 - def 함수이름(매개변수):
실행하고자 하는 함수내용
(변수, 상수, 연산자, 대입문, 선언문, 반복문, 조건문, 분기문, 함수호출 등)
 - 함수이름(매개변수)

프로그래밍 문법

- 변수: 데이터를 저장하는 공간
 - 다른 변수와 구별되게 고유한 이름을 붙인 것을 명칭
 - 이름을 붙이는 규칙(명칭 규칙)
 - 키워드 사용 안됨 (if, else, for, while, break, switch, case, True, False, and, or, not, def, return)
 - 대소문자 구분
 - 주로 알파벳, 숫자, 밑줄문자로 구성 (공백, +, - 기호 사용 안됨)
 - 첫 글자로 숫자 사용 안됨
 - 변수 사용
 - 파이썬의 경우 변수는 별도의 타입을 지정하지 않음
- ```
>>>testNum=100
>>>print(testNum)
100
>>>testStr="대한민국"
>>>print(testStr)
대한민국
```

# 프로그래밍 문법

- 자료형(타입) : 데이터(자료)를 표현하는 방법(형태(타입))가 있음)

- 수치형

- 정수형: 가장 간단한 수치형 (소수점이 없음)

- ```
>>>a=10
```

- ```
>>>print(a)
```

- 10

- 실수형: 소수점이 있는 수치형

- ```
>>>a=10.2e2
```

- ```
>>>print(a)
```

- 1020.0

- 복소수형: 실수부와 허수부가 있는 수치형

- ```
>>>a=1+2j
```

- ```
>>>b=2+4j
```

- ```
>>>print(a+b)
```

- (3+6j)

프로그래밍 문법

- 자료형(타입) : 데이터(자료)를 표현하는 방법(형태(타입))가 있음
 - 문자열: 문자의 나열로써 따옴표 안에 작성
 - 한글, 영문, 한자 모두 표현 가능
 - 따옴표 안에 작성하면 숫자로 문자열로 처리

```
>>> a="1"
>>> b="2"
>>> print(a+b)
12
```

프로그래밍 문법

- 자료형(타입) : 데이터(자료)를 표현하는 방법(형태(타입))가 있음)

- 문자열: 문자의 나열로써 따옴표 안에 작성

- 긴 문자열을 표현할 때는 따옴표 세 개를 사용

a="동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라 만세
무궁화 삼천리 화려 강산 대한 사람 대한으로 길이 보전 하세"

```
>>>print(a)
```

동해물과 백두산이 마르고 닳도록 하느님이 보우하사 우리나라 만세 무궁화 삼천리 화려 강산 대한
사람 대한으로 길이 보전 하세

```
>>>b="""남산 위에 저 소나무 철갑을 두른 듯 바람 서리 불변함은 우리 기상일세  
무궁화 삼천리 화려 강산 대한 사람 대한으로 길이 보전 하세"""
```

```
>>>print(b)
```

남산 위에 저 소나무 철갑을 두른 듯 바람 서리 불변함은 우리 기상일세
무궁화 삼천리 화려 강산 대한 사람 대한으로 길이 보전 하세

프로그래밍 문법

- 연산자: 프로그램의 산술식이나 연산식을 표현하고 처리하기 위해 제공되는 기호
 - 대입연산: 오른쪽에 있는 값을 왼쪽의 변수에 저장할 때 사용

```
>>>a=10
>>>print(a)
10
```
 - 산술연산: 숫자형 연산할 때 사용(사칙연산, 거듭제곱, 정수나누기, 나머지 연산)

```
>>>a=10
>>>b=a**2
100
>>>a=7
>>>b=a%2
```

프로그래밍 문법

- 연산자: 프로그램의 산술식이나 연산식을 표현하고 처리하기 위해 제공되는 기호
 - 복합대입연산: 오른쪽에 있는 변수와 값을 왼쪽 변수에 저장할 때 사용

```
>>>a=10
>>>a=a+20
>>>print(a)
30
```


프로그래밍 문법

- 자료형(타입)변환: 데이터(자료)를 표현하는 방법을 변환

- 문자열연산

```
>>>a="대한"
```

```
>>>b="민국"
```

```
>>>c="만세"
```

```
>>>print((a+b))+(c*3))
```

```
대한민국만세만세만세
```

- 정수형 변환: 문자열을 숫자(정수)로 변환하기 위해 int()함수를 사용

```
>>>a=10
```

```
>>>b="20"
```

```
print(a+(int(b)))
```

```
30
```

프로그래밍 설계

프로그래밍 명세

- 두 수를 입력 받아 더하고, 빼고, 곱하고, 나눈 결과를 출력하는 프로그램

프로그래밍 설계

프로그래밍 명세

- 두 수를 입력 받아 더하고, 빼고, 곱하고, 나눈 결과를 출력하는 프로그램

입력(Input)

- 두 수(값)
- 입력받는다

처리(Processing)

- 두 수(값)
- 더하고, 빼고, 곱하고, 나눈다

출력(Output)

- 처리 결과(값)
- 출력한다

프로그래밍 설계

프로그래밍 명세

- 두 수를 입력 받아 더하고, 빼고, 곱하고, 나눈 결과를 출력하는 프로그램

입력(Input)

- 두 수(값)
- 입력받는다

- 데이터:
- 변수(이름, 형태):
- 자료형:
- 연산자:
- 명령문:
- 함수:
- 이벤트:

처리(Processing)

- 두 수(값)
- 더하고, 빼고, 곱하고, 나눈다

- 데이터:
- 변수(이름, 형태):
- 자료형:
- 연산자:
- 명령문:
- 함수:
- 이벤트:

출력(Output)

- 처리 결과(값)
- 출력한다

- 데이터:
- 변수(이름, 형태):
- 자료형:
- 연산자:
- 명령문:
- 함수:
- 이벤트:

프로그래밍 설계

프로그래밍 명세

- 두 수를 입력 받아 더하고, 빼고, 곱하고, 나눈 결과를 출력하는 프로그램

입력(Input)

- 두 수(값)
- 입력받는다

- 데이터: 두 수
- 변수(이름, 형태): a,b(정수)
- 자료형: 정수형
- 연산자: 대입연산
- 명령문: 순서
- 함수: 입력함수(내장함수,사용자함수)
- 이벤트: ~할 때

처리(Processing)

- 두 수(값)
- 더하고, 빼고, 곱하고, 나눈다

- 데이터: 두 수, 처리 결과값
- 변수(이름, 형태): a,b,result(정수형)
- 자료형: 정수형
- 연산자: 대입, 사칙(=, -, *, /)연산
- 명령문: 순서
- 함수: 연산함수(내장함수,사용자함수)
- 이벤트: ~할 때

출력(Output)

- 처리 결과(값)
- 출력한다

- 데이터: 처리 결과값
- 변수(이름, 형태): result(정수형)
- 자료형: 정수형, 실수형
- 연산자: 대입
- 명령문: 순서
- 함수: 출력함수(내장함수,사용자함수)
- 이벤트: ~ 할 때

프로그래밍 설계

프로그래밍 명세

- 두 수를 입력 받아 더하고, 빼고, 곱하고, 나눈 결과를 출력하는 프로그램



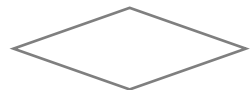
시작, 끝



초기값 선언(설정)



데이터 입,출력



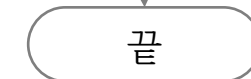
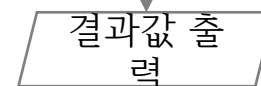
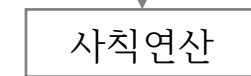
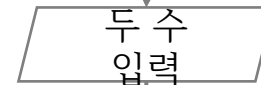
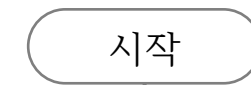
조건에 따른 판단(분기)



연산 및 처리



데이터 제어(흐름)



프로그래밍 맛보기

- 윈도우에서 파이썬을 설치
 - <http://www.python.org> 홈페이지에 접속
 - Downloads 메뉴 클릭 - 윈도우용 Python 설치 파일 내려받기(다운로드)
 - Download the latest version for Windows
 - Download Python 3.8.2
- 윈도우에서 파이썬을 실행
 - 시작 메뉴 항목에 오른쪽 클릭 보내기 -> 바탕화면에 바로 가기 만들기
- IDE (통합개발환경)
 - 다른 컴퓨터 환경에서도 같은 인터페이스로 작업 가능

프로그래밍 맛보기

- 파이썬
 - 파이썬 다운로드
 - <https://www.python.org/downloads/>
 - Download the latest version for Windows
 - Download Python 3.8.2
 - 파이썬 설치
 - Download Python 3.8.2
 - Install Now
 - 파이썬 IDLE (Python)
 - Python Shell
 - >>>

프로그래밍 맛보기

- 파이썬
 - 구글 코랩
 - <https://colab.research.google.com/>

프로그래밍 설계 – 실습1

프로그래밍 명세

- 이름과 출생연도를 입력 받아 이름과 현재 나이를 출력하는 프로그램

입력(Input)

■

처리(Processing)

■

출력(Output)

■

프로그래밍 설계 – 실습1

프로그래밍 명세

- 이름과 출생연도를 입력 받아 이름과 현재 나이를 출력하는 프로그램

입력(Input)

■

- 데이터:
- 변수(이름, 형태):
- 자료형:
- 연산자:
- 명령문:
- 함수:
- 이벤트:

처리(Processing)

■

- 데이터:
- 변수(이름, 형태):
- 자료형:
- 연산자:
- 명령문:
- 함수:
- 이벤트:

출력(Output)

■

- 데이터:
- 변수(이름, 형태):
- 자료형:
- 연산자:
- 명령문:
- 함수:
- 이벤트:

프로그래밍 설계 - 실습1

프로그래밍 명세

- 이름과 출생연도를 입력 받아 이름과 현재 나이를 출력하는 프로그램



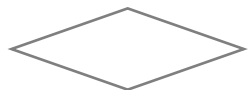
시작, 끝



초기값 선언(설정)



데이터 입,출력



조건에 따른 판단(분기)



연산 및 처리

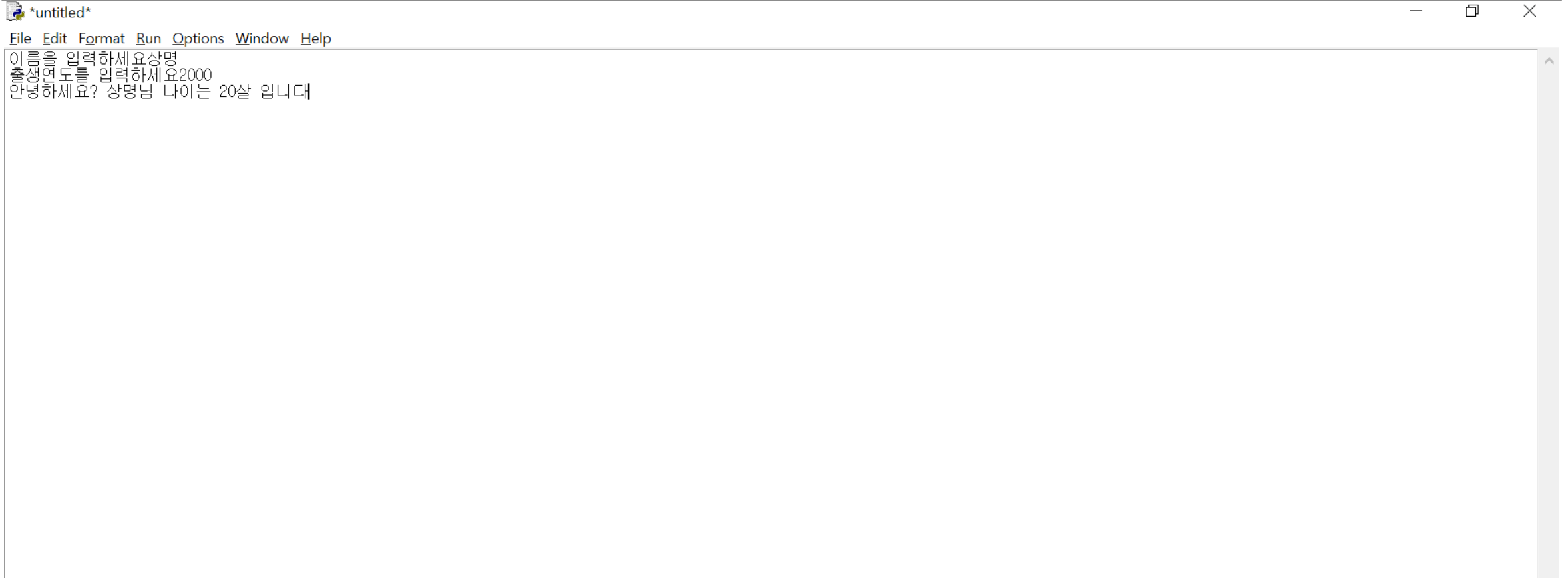


데이터 제어(흐름)

예제맛보기

프로그래밍 명세

- 이름과 출생연도를 입력 받아 이름과 현재 나이를 출력하는 프로그램



예제맛보기

프로그래밍 명세

- 1)사용자에게 ‘묵’, ‘찌’, ‘빠’ 문자열을 입력받고 입력된 문자열이 각 ‘묵’, ‘찌’, ‘빠’ 인 경우에만 각 해당 문자열을 출력한다
- 2)다른 문자열을 입력된 경우 “재입력” 메시지를 출력해준다
- 3)사용자가 ‘묵’, ‘찌’, ‘빠’ 각 문자열을 입력 하지 않고 엔터를 누르면 프로그램은 종료된다

untitled

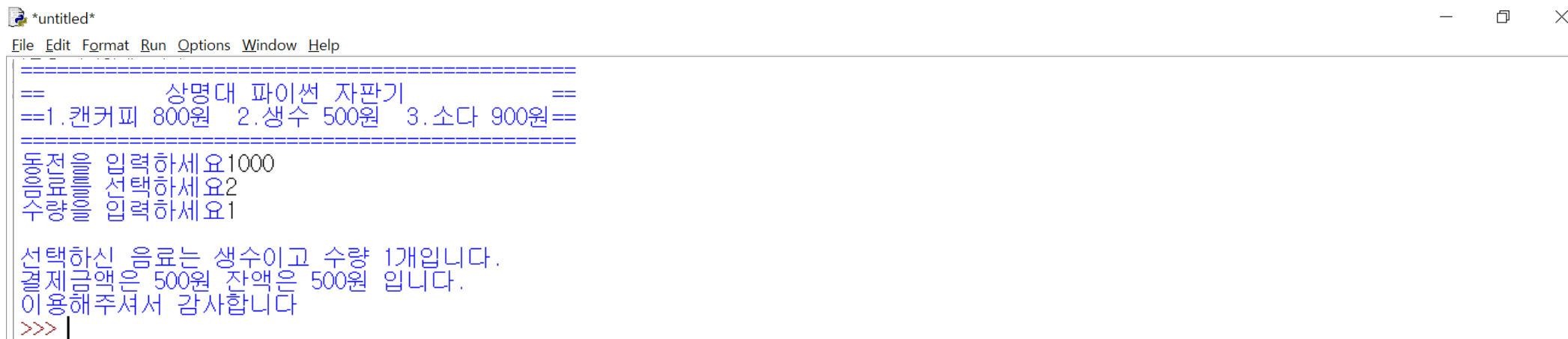
File Edit Format Run Options Window Help

```
묵 찌 빠 중 하나를 내주세요,종료하시려면 엔터를 눌러주세요묵
묵을 내셨습니다
묵 찌 빠 중 하나를 내주세요,종료하시려면 엔터를 눌러주세요찌
찌를 내셨습니다
묵 찌 빠 중 하나를 내주세요,종료하시려면 엔터를 눌러주세요빠
빠를 내셨습니다
묵 찌 빠 중 하나를 내주세요,종료하시려면 엔터를 눌러주세요가위
묵 찌 빠로 입력하세요
묵 찌 빠 중 하나를 내주세요,종료하시려면 엔터를 눌러주세요
종료합니다
>>> `
```

예제맛보기

프로그래밍 상세명세

- 1) 메뉴판을 출력하고 사용자로부터 투입 금액을 받고 투입금액이 최소금액(500원)보다 적으면 투입 금액 부족 문자열을 출력합니다
- 2) 투입금액이 최소금액보다 크면 사용자로부터 음료메뉴 선택과 수량을 입력 받습니다
- 3) 선택한 음료와 수량으로 결제 총액을 계산하고 총액이 투입금액보다 크거나 같으면 결제 금액이 부족하다는 문자열을 출력합니다
- 4) 결제 총액이 투입금액보다 적으면 거스름돈을 계산하고
- 5) 사용자에게 선택한 음료의 종류와 수량 결제금액과 잔액을 출력합니다



```
*untitled*
File Edit Format Run Options Window Help

=====
==          상명대 파이썬 자판기          ==
==1. 캔커피 800원  2. 생수 500원  3. 소다 900원==
=====

동전을 입력하세요1000
음료를 선택하세요2
수량을 입력하세요1

선택하신 음료는 생수이고 수량 1개입니다.
결제금액은 500원 잔액은 500원 입니다.
이용해주셔서 감사합니다
>>> |
```

프로그래밍 구현 - 실습1

맺음말