

컴퓨터 프로그래밍 II : Python

1주차 : 강의 오리엔테이션

조 현준 chris.cho
cho1475@korea.ac.kr



오픈톡방

Python 프로그래밍 II: 제2분반 대화방

<https://open.kakao.com/o/gDJEERag> (code: cose102)

목차

1. 강의 소개

- 1) 수업 계획 및 방식
- 2) 컴퓨터 프로그래밍의 정의
- 3) Python (파이썬) 소개 & 활용사례

2. 파이썬 개발 환경 구축 및 Hello World 출력

강의 계획 및 방식

주	수업내용	수업핵심질문 (핵심개념)	수업교재 및 자료	과제	활동 및 설계내용	총 16주차
1	파이썬 소개 및 개발환경구축	파이썬이 다른 프로그래밍 언어 와 어떻게 다른가요? (파이썬 언어 소개, 인터프리터 설치, jupyter notebook 사용법 등)	교재요약 PPT & 실습 노트북 파일 (*.ipynb)	개발 환경 설정 및 간단한 "Hello, World!" 프로그램 작성	강의 소개, 파이썬 소개 및 개발 환경 설정	
2	변수와자료형	파이썬에서 변수를 선언하고 사용하는 방법은 무엇인가요? (변수, 기본 자료형-정수, 실수, 불리언 등)	교재요약 PPT & 실습 노트북 파일 (*.ipynb)	온라인 저지 시스템을 사용한 코딩 테스트	자료형에 대한 이해도 향상을 위한 실습	
3	함수	함수는 왜 중요하며, 어떻게 정의 하나요? (함수 정의, 호출, 매개변수, 반환 값)	교재요약 PPT & 실습 노트북 파일 (*.ipynb)	온라인 저지 시스템을 사용한 코딩 테스트	함수를 활용한 문제 해결	
4	조건문	조건문은 프로그램에 어떤 영향을 미치나요? (if, else, elif 구문, 조건문의 논리)	교재요약 PPT & 실습 노트북 파일 (*.ipynb)	온라인 저지 시스템을 사용한 코딩 테스트	조건에 따른 다양한 흐름 제어	
5	반복문	반복문은 어떻게 사용하며, 왜 중요한가요? (for, while 반복문, 반복 제어)	교재요약 PPT & 실습 노트북 파일 (*.ipynb)	온라인 저지 시스템을 사용한 코딩 테스트	반복문을 이용한 복잡한 문제 해결	
6	문자열	파이썬에서 문자열을 어떻게 다루나요? (문자열 처리, 메소드, 슬라이싱)	교재요약 PPT & 실습 노트북 파일 (*.ipynb)	온라인 저지 시스템을 사용한 코딩 테스트	문자열 기반 데이터 처리 실습	
7	특강: 오픈소스, ChatGPT란? (+실무 개발자 경험 공유)	실무 개발에서 오픈소스와 AI란 무엇인가요? (오픈소스 기여, ChatGPT 활용, 실무 경험)	교재요약 PPT	X	특강 참여 및 토론	
8	중간고사	지금까지 배운 내용에 대한 평가		X	시험 준비 및 학기 내용 복습 및 코딩 테스트 시험	

강의 계획 및 방식

주	수업내용	수업핵심질문 (핵심개념)	수업교재 및 자료	과제	활동 및 설계내용
9	자료구조1 : 리스트, 튜플	리스트와 튜플은 어떻게 다르며, 어떤 상황에서 사용하는 것이 좋나요? (리스트와 튜플의 정의, 사용법, 차이점)	교재요약 PPT &실습 노트북 파일 (*.ipynb)	온라인 저지 시스템을 사용한 코딩 테스트	다양한 자료구조 활용 실습
10	자료구조2: 셋, 딕셔너리	셋과 딕셔너리는 어떤 경우에 사용되며, 어떤 장점이 있나요? (셋과 딕셔너리의 특징, 사용법)	교재요약 PPT &실습 노트북 파일 (*.ipynb)	온라인 저지 시스템을 사용한 코딩 테스트	효율적인 데이터 구조 사용 방법 탐색
11	파일 입출력	파이썬에서 파일 입출력을 처리하는 방법은 무엇인가요? (표준 입출력, TXT, JPG, CSV, JSON 파일 읽기, 쓰기)	교재요약 PPT &실습 노트북 파일 (*.ipynb)	X	파일 시스템과 상호작용하는 프로그래밍 실습
12	데이터베이스	데이터베이스를 파이썬과 어떻게 연결하며, 왜 중요한가요? (데이터베이스 기초, SQL, 파이썬에서의 데이터베이스 연동)	교재요약 PPT &실습 노트북 파일 (*.ipynb)	X	데이터베이스 기반 프로그래밍 실습
13	클래스	클래스와 객체 지향 프로그래밍의 장점은 무엇인가요? (객체 지향 프로그래밍, 클래스와 객체, 상속과 다형성)	교재요약 PPT &실습 노트북 파일 (*.ipynb)	X	객체 지향적 접근 방식을 통한 문제 해결
14	테스팅	테스트 자동화란 무엇인가요? (소프트웨어 테스팅, 테스트 자동화, 단위 테스트)	교재요약 PPT &실습 노트북 파일 (*.ipynb)	X	테스팅 개념과 테스트 자동화, 테스트 주도 개발(TDD) 방법론 적용
15	예외처리	예외 처리가 중요한 이유는 무엇이며, 어떻게 구현하나요? (예외와 에러, 예외 처리, 안정적인 코드 작성)	교재요약 PPT &실습 노트북 파일 (*.ipynb)	X	안정적이고 신뢰성 있는 코드 작성 실습
16	기말고사	전체 학기 내용에 대한 종합 평가		X	시험 준비 및 학기 내용 복습 및 코딩 테스트 시험

총 16주차

주 1회 3시간 :

이론 + 실습 + Q&A

- 1시간 15분 : 1부(이론 + 실습)

- 15분 : 쉬는 시간

- 1시간 15분 : 2부(실습 + Q&A)

강의 계획 및 방식

총16주차

주 1회 3시간
인증 시스

Q&A

1부(이론 + 실습) 간 2부(실습 + Q&A)

현업에서 활용될 수 있는 다양한 실무 코딩 및 현업 개발자의 노하우

평가 방식

절대 평가(100점 만점 기준)

1. 중간고사: 25점 - 코딩 테스트

2. 기말고사: 25점 - 코딩 테스트

3. 과제: 30점 - 실습 예제 및 실습 과제

1. 실습 예제: 수업 시간 중 실시간으로 진행, 완료 못할 경우 다음 수업 전 제출

2. 실습 과제: 수업 외 개별 작업, 총 2회 제출

4. 출석: 10점

1. 1번 결석 시 1점 감점, 5번 이상 결석 시 F등급 부여

2. 수업 시작 후 30분 이상 지각 시 지각 처리, 지각 3회는 결석 1회로 간주

5. 참여도: 10점

1. 수업 중 질문이나 답변에 참여한 횟수(오픈 톡 포함)를 기준으로 점수 산정

2. 만점 기준: 16회 이상 참여

3. 예: 참여 횟수가 8회일 경우, 참여도 점수 50% 비율 적용하여 5점 부여

왜 우리는
컴퓨터 프로그래밍 과목을
수강해야 하는가??



컴퓨터 프로그래밍의 정의

컴퓨터 프로그램을 만드는 행위(기술)

컴퓨터란? (입출력 요소를 가진) 계산 기능이 있는 전자 기기를 총칭



프로그램이란? 진행 계획, 절차, 순서

```
var scrollElement = document.querySelector('div');
element.clientHeight + 0.02 * window.innerHeight;
window.scroll(0, scrollHeight);
```

출처 <https://giphy.com/gifs/screen-monitor-closeup-26tn33aiTi1jkl6H6>

컴퓨터 프로그램이란? 컴퓨터에서 절차적으로 실행되는 명령어(코드) 집합

컴퓨터 프로그래밍을 배우는 이유 1

사람이 하는 일 (혹은 특정 문제)을 컴퓨터가 대신 수행 하는 것 → (반)자동화



(a) 아마존 고



(b) 아마존 키바

스탠퍼드대-구글 딥마인드, '모바일 알로하' 개발
요리·세탁·청소에다 반려동물 놀아주기까지 척척
기존 가사 보조·지원 로봇들보다 비용도 저렴해



(c) 무인지하철



(d) 무인자동차



(사진=Zipeng Fu 유튜브 채널)

[일상생활 속 프로그래밍의 결과물]

컴퓨터 프로그래밍을 배우는 이유 2

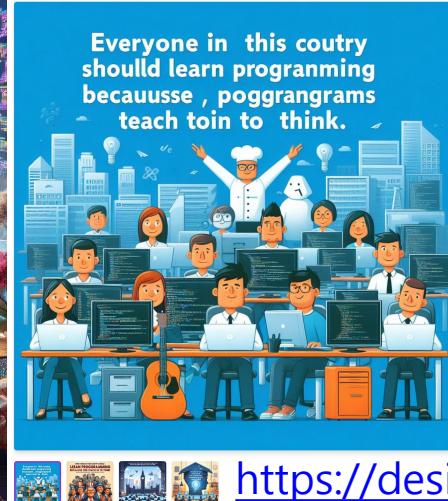
하나의 기술이나 학문으로서 프로그래밍 언어를 배우는 것이 아니라,
논리적인 사고를 훈련하기 위해 프로그래밍 언어를 배워야 한다

"이 나라의 모든 사람이 프로그램을 배워야 한다. 왜냐하면 프로그램이라는 것은 생각을 어떻게 하
는지 가르쳐 주기 때문이다."

- 스티브 잡스



ChatGPT4.0



이 나라의 모든 사람은 프로그래밍을 배워야 한다. 왜냐하면 프로그램이라는 것은 생
각을 어떻게 하는지 가르쳐주기 때문이다 -스티브 잡스-

[Download](#) [Copy](#) [Create design](#) [Edit image](#)

[Report image](#)

Content credentials ⓘ
Generated with AI - February 4, 2024 at 11:34pm



<https://designer.microsoft.com/image-creator>

파이썬 소개



파이썬 소개

파이썬의 등장

- 파이썬(Python)은 귀도 반 로섬(Guido Van Rossum) 1991년에 개발한 언어
- 처음에는 C 언어 기반으로 개발되었는데 이후 다양한 기능이 개발되어 추가됨



[귀도 반 로섬]



[파이썬 로고]

파이썬 소개

파이썬을 배우는 이유

1. 쉽고 간단한 프로그래밍 언어

- 화면에 'Hello World!'라는 텍스트를 출력하는 프로그램을 작성한다고 가정하자.
파이썬과 자바로 코드를 작성하면 아래와 같다.

파이썬

```
print("Hello World!")
```

자바(Java)

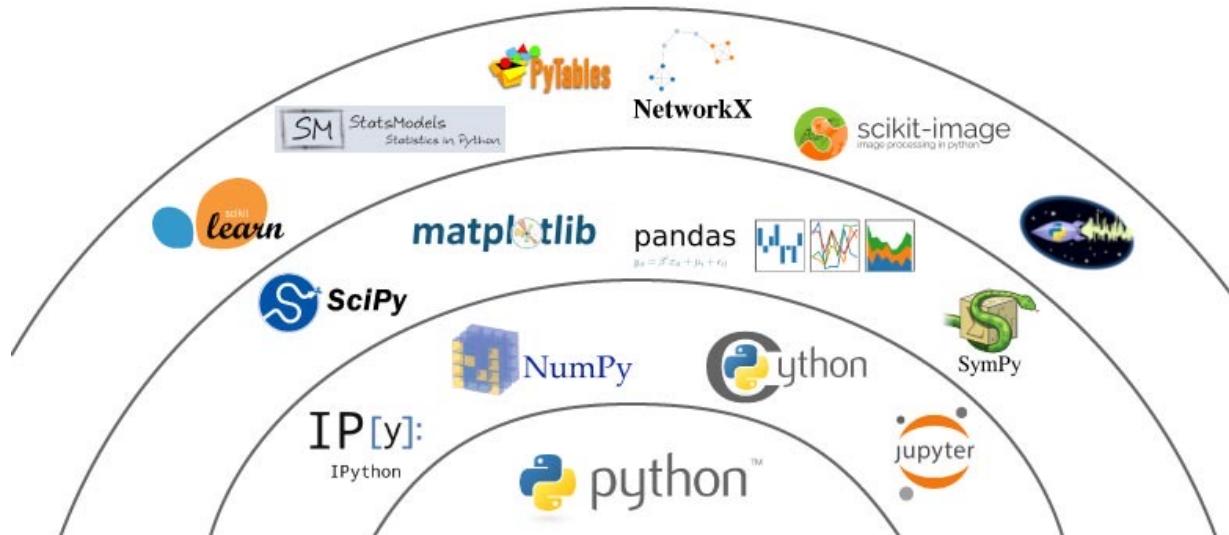
```
class HelloWorldApp {  
    public static void main(String[] args) {  
        System.out.println("Hello World!");  
    }  
}
```

파이썬 소개

파이썬을 배우는 이유

2. 다양한 라이브러리 제공

- 파이썬은 다양한 라이브러리를 제공하여 활용 범위가 넓다.
- 개발자들이 만들어 놓은 많은 모듈과 패키지를 쉽게 사용할 수 있다.



[Python Libraries] 출처 : <https://www.linkedin.com/pulse/top-10-python-libraries-pranshu-jaryal/>

파이썬 소개

파이썬을 배우는 이유

3. 대중적인 프로그래밍 언어

Rank	Jan 2024	Jan 2023	Change	Programming Language	Ratings
1	1	1		 Python	13.97%
2	3	3		2023, 2024년 프로그래밍 언어 랭킹 1순위 : 파이썬	
4	4			Java	7.87%
5	5			C#	7.16%
6	7	7		JavaScript	2.77%
7	10	10		PHP	1.79%
8	6	6		Visual Basic	1.60%
9	8	8		SQL	1.46%
10	20			Scratch	1.44%

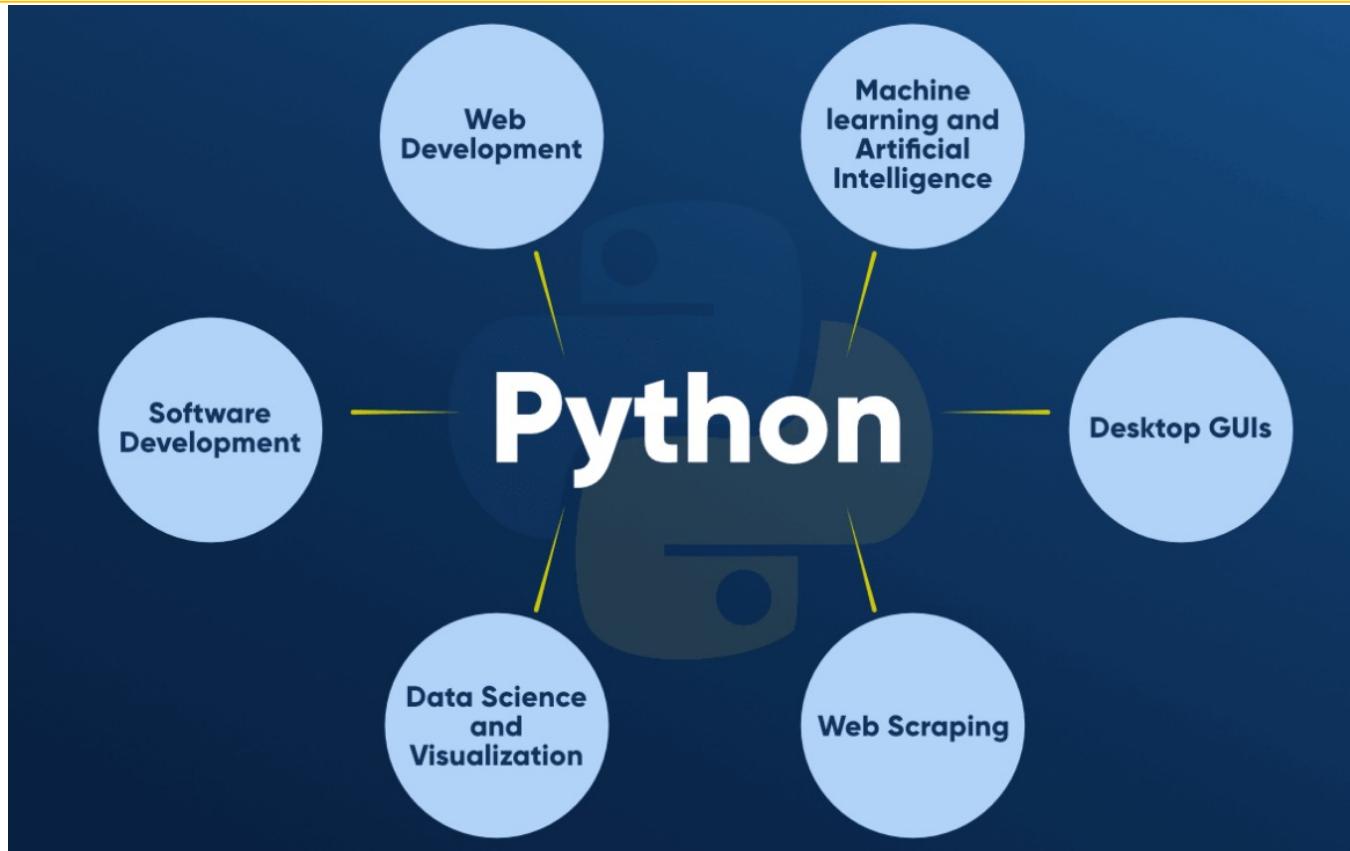
2024년 TIOBE Index 프로그래밍 인기 지수 결과

파이썬 소개

파이썬의 특징

- 플랫폼 독립적인 언어** : 어떤 운영체제든 상관없이 사용할 수 있는 언어
- 인터프리터 언어** : 컴파일러 언어와 달리, 소스코드 자체가 바로 실행되는 특징이 있는 언어이다. 이로 인해 속도는 느리지만, 굉장히 간편하게 사용할 수 있음
- 객체 지향 언어** : 해당 프로그램이 해결해야 할 문제의 구성요소를 요소별로 정의한 뒤 각 요소의 기능(메서드)과 정보(속성)를 정의하여 요소들을 결합하고, 프로그램을 작성하는 방식
- 동적 타이핑 언어** : 프로그램의 실행 시점에 각 프로그램 변수의 타입을 결정하는 언어

파이썬 활용 영역



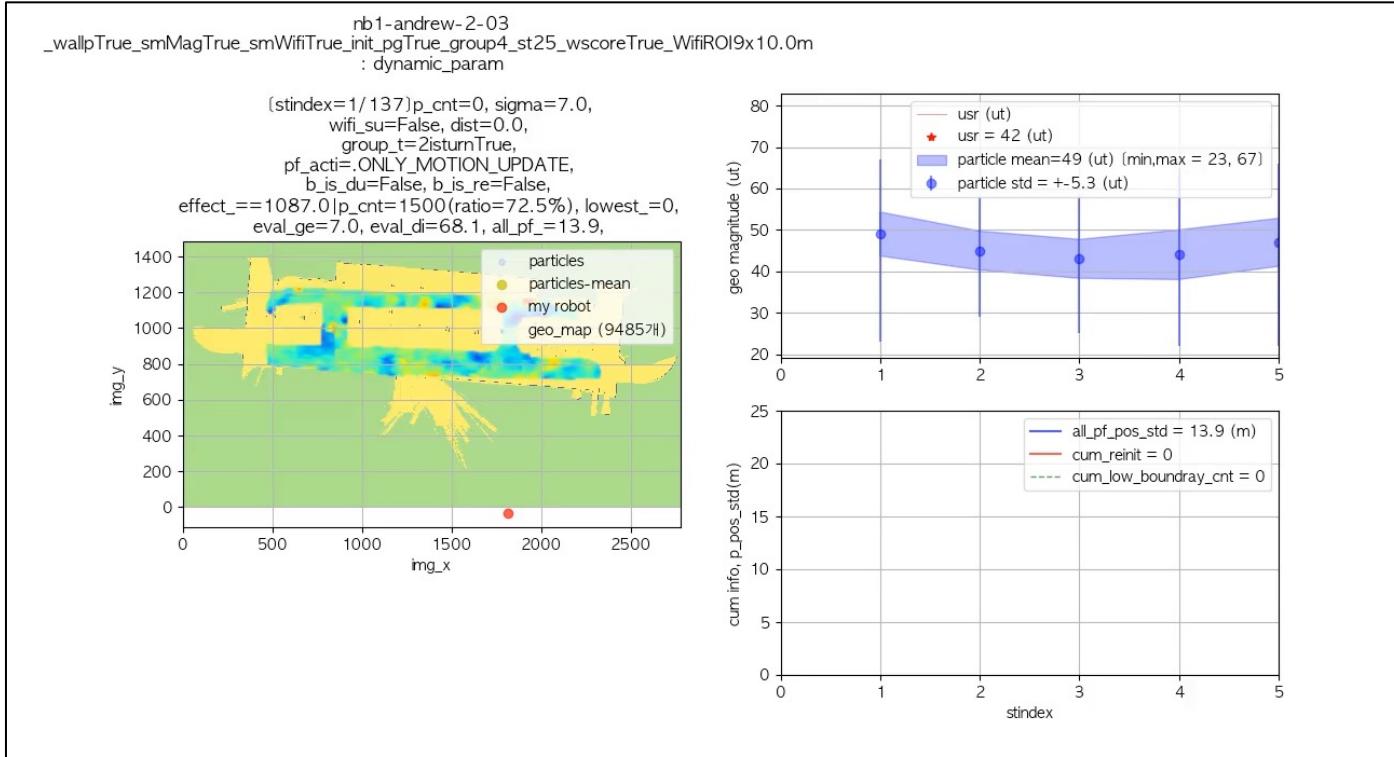
출처 : <https://medium.com/analytics-vidhya/python-roadmap-to-follow-in-2021-1f49715468ca>

파이썬 활용 사례



실내 로드뷰 관련 사례 - SLAM&실내 내비게이션(요약)

필요한 알고리즘(SLAM을 활용한 측위, 멀티 센서 융합 기반 Particle Filter를 활용한 위치 추정, etc)



1부 끝 10분 휴식

쉬면서 간단한 설문 조사 부탁 드려요
<https://forms.gle/EZCwVLsDicBaXE1y8>



파이썬 개발 환경 구축 및 설치



파이썬 개발 환경과 설치

파이썬 개발 환경 설정

1. 운영체제 선정

운영체제	장점	단점
윈도(Windows)	국내 사용자가 쓰기에 가장 쉬운 운영체제로, 프로그래밍을 시작하는 사람에게 상당히 좋은 선택임	다른 사용자가 개발한 다양한 모듈을 설치하기 어렵고, 참고 문서가 부족함
리눅스(Linux)	모듈 설치가 매우 쉽고, 다양한 레퍼런스를 인터넷에서 검색하기 쉬움	초기 사용자가 사용하기에는 운영체제 자체가 매우 어려움
맥 OS(Mac OS)	기본적으로 리눅스의 장점이 있으면서, 윈도처럼 안정적임	다른 운영체제보다 상대적으로 가격이 비쌈

[운영체제별 특징]

파이썬 개발 환경과 설치

파이썬 개발 환경 설정

2. 파이썬 인터프리터 선정

종류	설명
Python	일반적인 파이썬, 기본적인 모듈 포함
Anaconda	다양한 과학 계산용 모듈을 묶어 패키지로 제공, 이 책에서 사용하는 툴
Canopy	다양한 과학 계산용 모듈을 묶어 패키지로 제공

[주요 파이썬 인터프리터]

파이썬 개발 환경과 설치

파이썬 개발 환경 설정

3. 코드 편집기 선정

종류	설명
메모장	윈도의 기본 문서 편집 도구
VI editor	리눅스의 기본 문서 편집 도구
Jupyter Notebook	인터랙티브 형식의 라이브 코드를 제공하는 웹 베이스 편집 도구 <u>(인터넷 상관 없이 동작)</u>
PyCharm	다양한 기능을 갖춘 파이썬 전용 개발 도구(IDE)
Google Colab	Jupyter Notebook 기반의 구글에서 만든 웹 편집 도구 (가입만 하면 됨, 인터넷 연결이 되야 동작)

파이썬 개발 환경과 설치 (Jupyter 설치 + python 3.x)

1. 실습 가이드 : [실습가이드 URL](#)

Python 실습 관련 가이드

1. 윈도우 OS환경 기반, 아나콘다용 주피터 노트북을 활용시

문제가 있을 경우 colab을 활용하여 실습해주시기를 바랍니다.

2. extension 설치는 실습 (필수 사양이 아닌) 권장사항입니다. extension 미설치의 경우,
실습을 진행하는데 지장 없습니다.

3. Contest 비밀번호는 102 입니다.

4. 가이드 내용 외에 추가 문의는 이메일(cho1475@korea.ac.kr)로 회신 바랍니다.

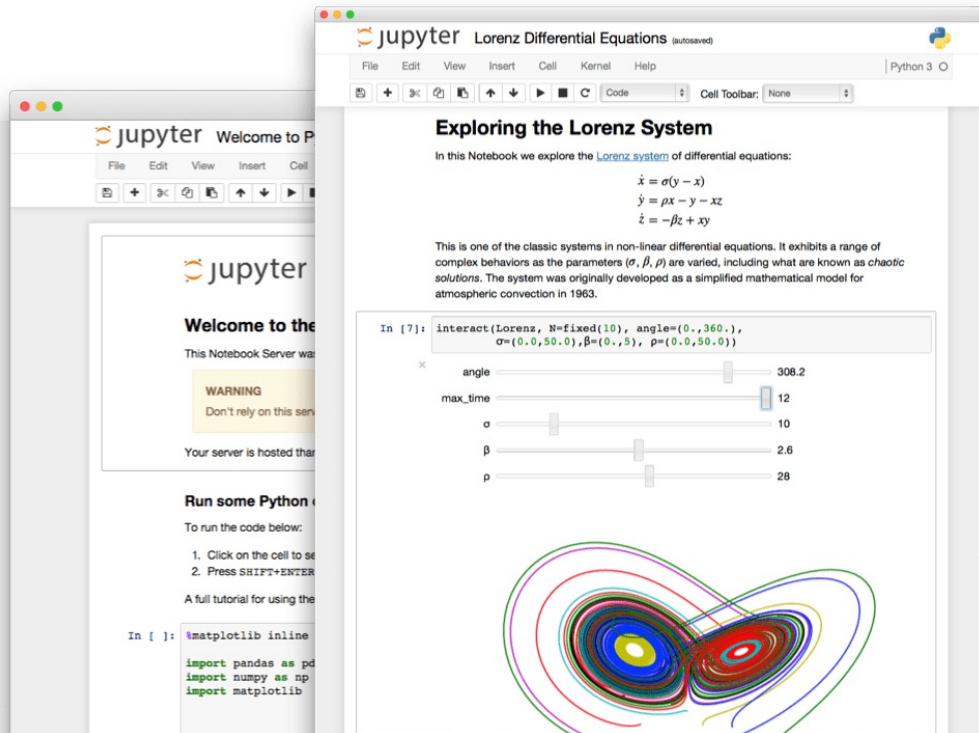
[리스트 보기](#)

파이썬 프로그래밍 설치 및 실습 관련 가이드

[아나콘다 기반] jupyter notebook 설치 가이드(FAQ, Extension 설치)

온라인 저지 홈페이지 사용 가이드

실습 자료(.ipynb) 읽어오는 방법



[Jupyter Notebook 실행화면]

파이썬 개발 환경과 설치 (그외 설치 방법 2가지)

2. (Option, 고급자용) Pycharm IDE 개발 환경 구축 :

설치 파일 실행(아나콘다 포함)

<https://www.jetbrains.com/ko-kr/lp/pycharm-anaconda/>

3. (제일 쉬운 방법)온라인 GDB :

홈페이지 접속 → Language 'Python3' 선택

<https://www.onlinegdb.com/>

vs code 써도 괜찮습니다!!!

온라인 저지(자체 코딩 문제 홈페이지)

1. 실습 가이드 : [실습가이드 URL](#)

Python 실습 관련 가이드

1. 윈도우 OS환경 기반, 아나콘다용 주피터 노트북을 활용시
문제가 있을 경우 colab을 활용하여 실습해주시기를 바랍니다.

2. extension 설치는 실습 (필수 사양이 아닌) 권장사항입니다. extension 미설치의 경우,
실습을 진행하는데 지장 없습니다.

3. Contest 비밀번호는 102 입니다.

4. 가이드 내용 외에 추가 문의는 이메일(cho1475@korea.ac.kr)로 회신 바랍니다.

[리스트 보기](#)

파이썬 프로그래밍 설치 및 실습 관련 가이드

 [아나콘다 기반] jupyter notebook 설치 가이드(FAQ, Extension 설...)

 온라인 저지 홈페이지 사용 가이드

 실습 자료(.ipynb) 읽어오는 방법

자동 채점 결과 : Accept or Wrong
(실습/과제 및 중간/기말 고사 활용 예정)

<http://oj.rightline.kr/contest>

- ID : 학번 (*점수 반영 기준)

온라인 저지(무료 상용 코딩 문제 홈페이지)

1. 프로그래머스 : <https://school.programmers.co.kr/>

2. 해커랭크 : <https://www.hackerrank.com/>

3. 백준 : <https://www.acmicpc.net/>

The screenshot shows the Baekjoon Online Judge interface. At the top, there are navigation links: '제출', '맞힌 사람', '숏코딩', '제재점 결과', '제한 현용', '내 제출', '난이도 기여', and '질문 검색'. Below this, a progress bar indicates '2439번' solved out of '2439번' total. The main content area is titled '별 찍기 - 2' with a '보기' button. It includes a table with columns: 시간 제한, 메모리 제한, 제출, 정답, 맞힌 사람, and 정답 비율. The first row shows values: 1 초, 128 MB, 212800, 118952, 101919, and 56.387%. Below the table, there's a section titled '문제' with the following text:

첫째 줄에는 별 1개, 둘째 줄에는 별 2개, N번째 줄에는 별 N개를 찍는 문제
하지만, 오른쪽을 기준으로 정렬한 별(예제 참고)을 출력하시오.

입력

첫째 줄에 N ($1 \leq N \leq 100$)이 주어진다.

출력

첫째 줄부터 N 번째 줄까지 차례대로 별을 출력한다.

예제 입력 1 복사

```
5
```

예제 출력 1 복사

```
*
**
***
*****
*****
```

The screenshot shows the HackerRank platform. At the top, it says 'HackerRank' and 'Introduction Your First Challenge'. The challenge description is: 'This is an introductory challenge to help you get familiar with the HackerRank coding environment. Complete the function solveMeFirst to compute the sum of two integers.' The 'Function prototype:' is given as: `int solveMeFirst(int a, int b);`. The 'Return values' are: '• sum of the above two integers'. The 'Sample Input' is: `a = 2
b = 3`. The 'Sample Output' is: `5`. The 'Explanation' is: 'None'. On the right, there is a code editor window with the following Java code:

```
1 // #include <stdio.h>
2 // #include <string.h>
3 // #include <math.h>
4 // #include <stdlib.h>
5
6 * int solveMeFirst(int a, int b) {
7     // Hint: Type return a+b; below
8
9 }
10
11 * int main() {
12     int num1,num2;
13     scanf("%d %d",&num1,&num2);
14     int sum;
15     sum = solveMeFirst(num1,num2);
16     printf("%d",sum);
17     return 0;
18 }
```

Below the code editor, there are buttons for 'Run Code' and 'Line: 1 Col: 1'. A status bar at the bottom right says '종 후 제출하기'.

감사합니다.

본 강의 교안은 한빛 아카데미(주)에서 제공된
강의 교안을 참고하여 제작되었습니다.

