



# Introduction to this course

---

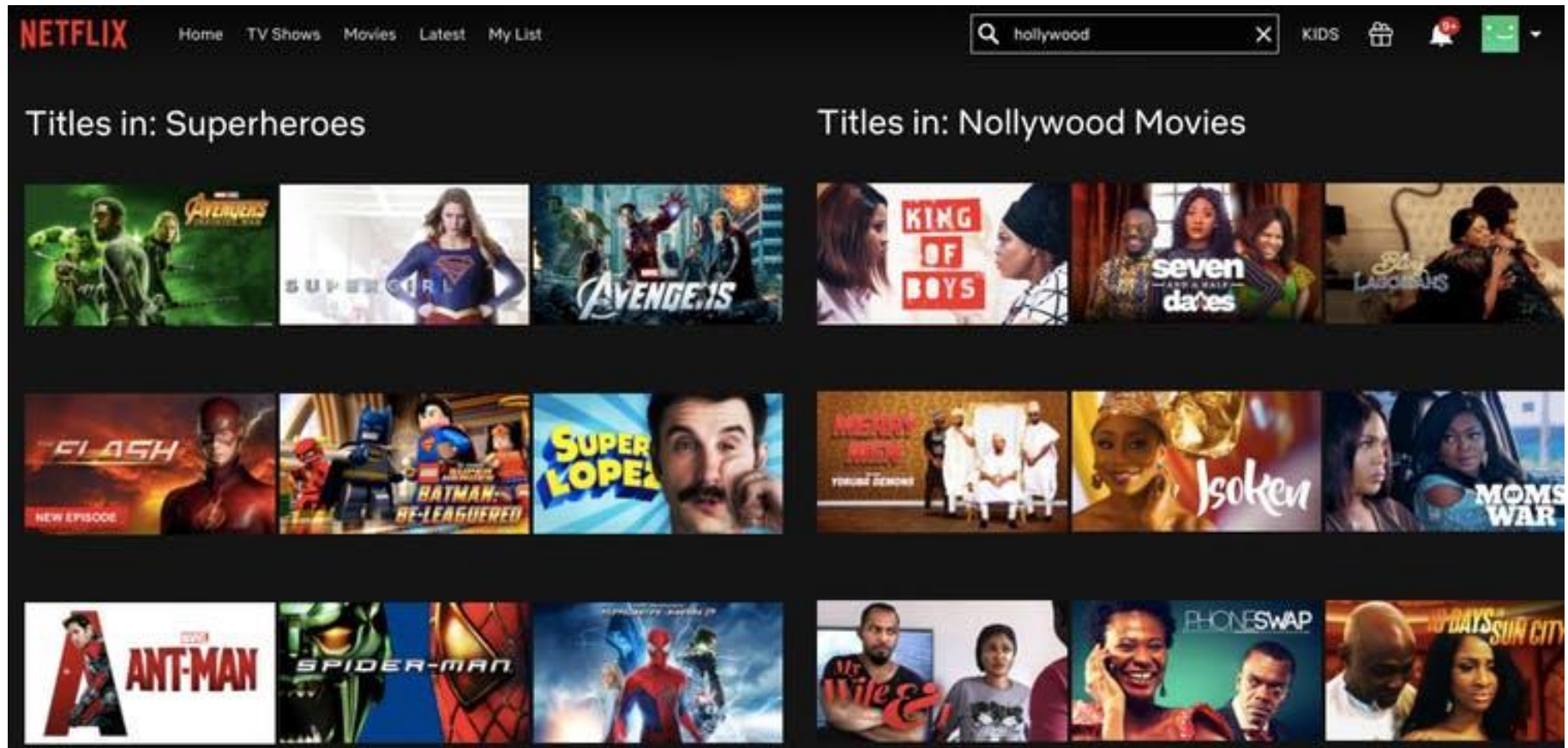
천세진

2021 Spring

# 컴퓨터 프로그램(Computer Program)이란?

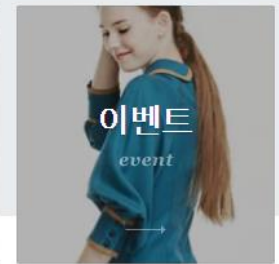
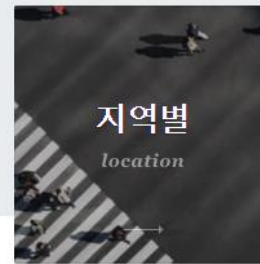
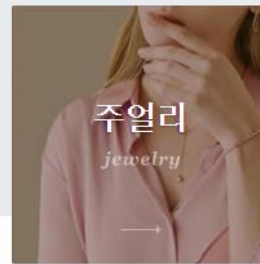
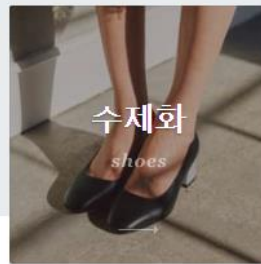
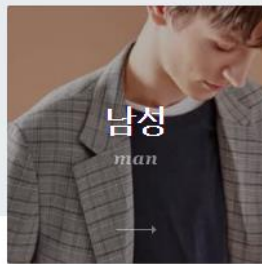
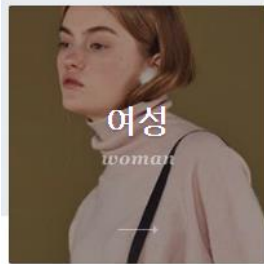


# 컴퓨터 프로그램(Computer Program)이란?





# 컴퓨터 프로그램(Computer Program)이란?



## Today

오늘 인기 상품 · 코디 추천 ⓘ



인기 전체



베네노



모노타임



메이퍼블



드레스빌



오브제스트



비헤이즐



바이름



어쓰



오조이



market



980 원

여성 골지 벗겨지지 않는 갑종 덧신



18,900 원

[무배] 1+1 휘두루 루즈핏 반팔 티셔츠 14color



39,000 원

[무배] 우아한 드레이프 보트넥 플레어 원피스



13,060 원

[무배] 투버튼 세미 보이핏 데님팬츠



19,800 원

[오늘출발] 세트 set/레이어드 후드 오버핏 티셔츠+조거 밴딩 팬츠 세트



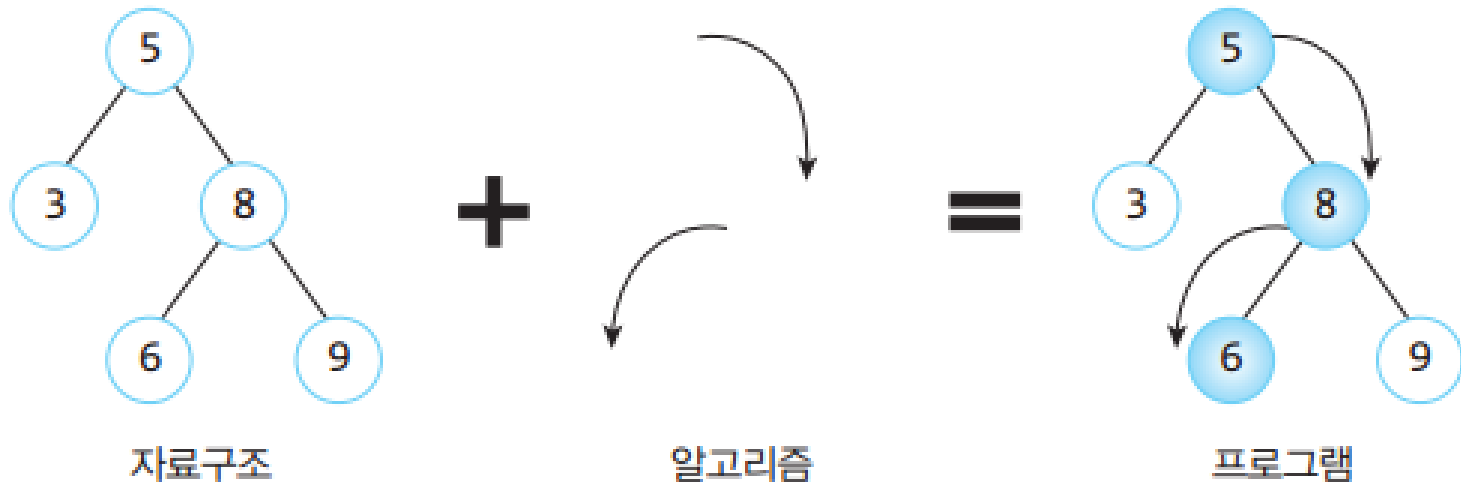
#롱서츠 #캠퍼스룩

#봄신상코디



# 컴퓨터 프로그램(Computer Program)이란?

## ■ 자료구조 + 알고리즘

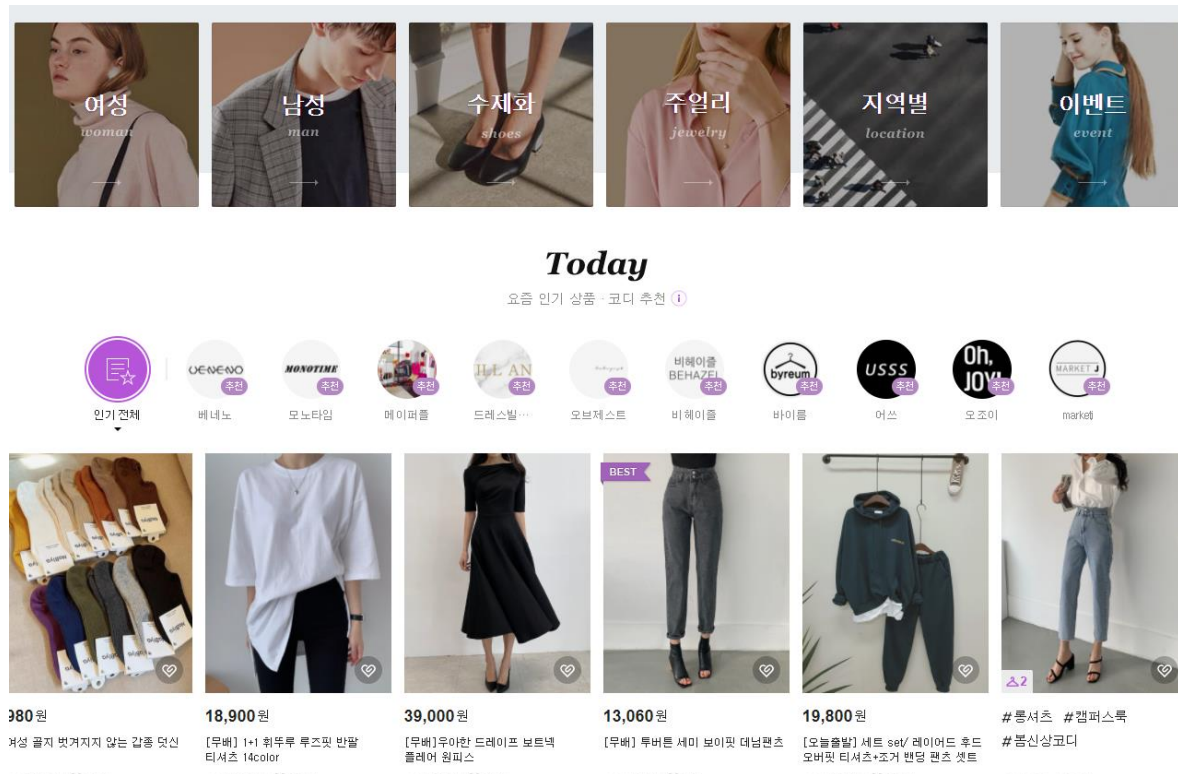


# 자료구조

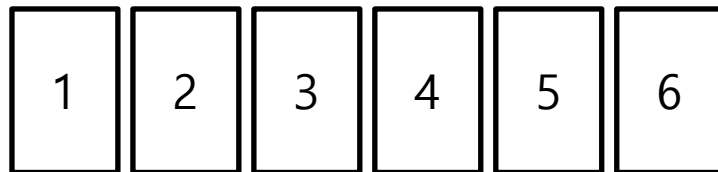
- 일상 생활에서 자료를 정리하고 조직화하는 이유는?
  - 사물을 편리하고 효율적으로 사용하기 위함
  - 다양한 자료를 효율적인 규칙에 따라 정리한 예



# 컴퓨터 프로그램(Computer Program)이란?



## 자료구조



Item1 ~ 6

## 알고리즘

"좋아요"의 수가 많은 순으로 정렬한다





## ■ 컴퓨터과학 전문가들과 의사소통하기





# 왜 배워야 하는가? (2)

## ■ 코딩테스트의 기본 지식

Coding Test Practice > 정렬 > K번째수

Help Compile option

dark light sublime vim emacs Python3

K번째수

Description

배열 array의 i번째 숫자부터 j번째 숫자까지 자르고 정렬했을 때, k번째에 있는 수를 구하려 합니다.

예를 들어 array가 [1, 5, 2, 6, 3, 7, 4], i = 2, j = 5, k = 3이라면

1. array의 2번째부터 5번째까지 자르면 [5, 2, 6, 3]입니다.
2. 1에서 나온 배열을 정렬하면 [2, 3, 5, 6]입니다.
3. 2에서 나온 배열의 3번째 숫자는 5입니다.

배열 array, [i, j, k]를 원소로 가진 2차원 배열 commands가 매개변수로 주어질 때, commands의 모든 원소에 대해 앞서 설명한 연산을 적용했을 때 나온 결과를 배열에 담아 return 하도록 solution 함수를 작성해주세요.

제한사항

- array의 길이는 1 이상 100 이하입니다.
- array의 각 원소는 1 이상 100 이하입니다.
- commands의 길이는 1 이상 50 이하입니다.
- commands의 각 원소는 길이가 3입니다.

입출력 예

array	commands	return
[1, 5, 2, 6, 3, 7, 4]	[[2, 5, 3], [4, 4, 1], [1, 7, 3]]	[5, 6, 3]

입출력 예 설명

solution.py

```
1 def solution(array, commands):
2     answer = []
3     return answer
```

Result

Result of [Run] or [Submit] will be displayed here

# 왜 배워야 하는가? (3)

- 더 명확하게 분석, 비교할 수 있는 능력을 함양
  - 이제는 better라는 말보다는 faster, cheaper, ..



<source: <http://www.mediasr.co.kr/news/articleView.html?idxno=48473>>

<source: <http://biz.newdaily.co.kr/site/data/html/2019/07/26/2019072600081.html>>



# 환영합니다

---

- 15주 동안, 기본적인 데이터구조와 알고리즘을 배울 것입니다
  - 전통적인 데이터구조/알고리즘
  - 그들의 효율성을 엄격하게 분석하는 방법
  - 언제 그것들을 사용해야할 지
  - 큐, 딕셔너리, 그래프, 정렬 등
  
- 오늘은 ...
  - 소개 및 강의 구조
  - 파이썬에 대한 기초 소개



# 커뮤니케이션

---

- 우선, 이메일로만 sjchun@dau.ac.kr
  - 강의, 프로젝트 관련 모든 주제
  - 개별 면담
  - 통상 3일 내로 답변 가능
  
- 면담가능 시간
  - 이메일로 사전 약속
  - 화, 수, 금 (AM 09:00~ PM 06:00)





# 강의 자료

---

## ■ Slides

- [sejinchun.com](http://sejinchun.com) > lectures

## ■ Textbook

- 파이썬과 함께하는 자료구조의 이해(Data structures with Python) - 양성봉
- Data Structures and Algorithms Using Python



# 과제 및 시험

---

## ■ 5개의 개별 과제 (30%)

- 대부분 프로그래밍 숙제이나, 쓰기 숙제도 있음
- Google Colab위에서 개별적으로 작성 후, 한번에 최종 제출
- 단순 코딩보다는 “상위레벨 개념 모델링”이 중요함
- 첫 숙제는 오늘부터



# 협력과 학업의 진정성 (1)

- 모든 프로젝트는 다음 사항을 준수해야 합니다
  - 어떠한 곳의 소스를 직접적으로 복사하지 마세요
  - 다른 코드의 소스를 참고하였다면, 꼭 프로젝트 내 참조사항을 표기하세요
  - 파일을 직접적으로 공유하지 말기 바랍니다.



# 협력과 학업의 진정성 (1)

- 프로젝트에 대한 협력은 다음과 같이:
  - 토의
  - 상위 수준의 pseudocode
  - 디버깅 도움
- 모든 협력에 대해서
  - 사용자 공유 등록 필수 (최대 2인)
  - 코드 내의 공유된 사용자 표기





# 이 강의에서 큰 성취를 하는 방법

---

- 수업에 제때 온다
  - 첫 5분이 마지막 5분보다 중요하다
- 수업 시간동안 노트북은 필요없다





# Abstract Data Type(ADT) & HW#1

---

추상 자료형(ADT)

# 추상 자료형 (ADT: Abstract Data Type)



# 컴퓨터 프로그램상의 추상화

## ■ 추상화(abstraction)의 개념

- 주어진 작업이나 객체를 속성들의 일부분을 가지고 필요한 만큼만 묘사할 수 있는 방법을 지원하는 것
- 필수적인 속성만으로 주어진 것을 묘사하여 나머지 속성들은 추상화, 은닉(information hiding), 또는 삭제됨

## ■ 추상화의 범주

- 데이터 추상화
  - 문자열, 수, 트리와 같은 계산될 자료의 특성을 추상화
- 제어 추상화 (알고리즘 추상화)
  - 실행 순서를 제어하는 특성을 추상화
  - 제어 추상화의 예: 반복문, 조건문, 프로시저 호출 등





# 데이터 추상화(data abstraction)

## ■ 처리할 자료, 연산, 자료형에 대한 추상화 표현

Person
firstName = "David"
lastName = "Wolber"
id = "922-43-9873"
email = "wolber@usfca.edu"
friends = []

```
2 class Person:
3     def __init__(self):
4         self.firstName = ""
5         self.lastName = ""
6         self.id = ""
7         self.email = ""
8         self.friends = []
9
10
11 p = Person()
12 p.firstName = "Sejin"
13 p.lastName = "Chun"
14 p.id = "111111"
15 p.email = "sjchun@dau.ac.kr"
16 p.friends.append("Hong Gil-dong")
```

# 제어 추상화(Procedural abstraction)

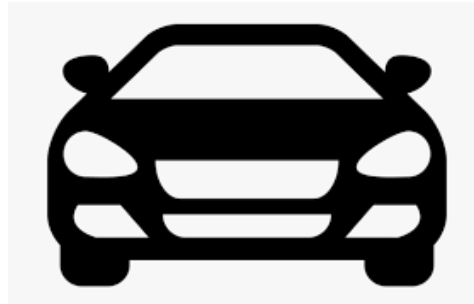
## ■ 처리할 함수나 메소드에 대한 추상화 표현

```
2  class Person:
3      def __init__(self):
4          self.firstName = ""
5          self.lastName = ""
6          self.id = ""
7          self.email = ""
8          self.friends = []
9
10     def update_username(self, first, last):
11         self.firstName = first
12         self.lastName = last
```

**\*사용자 이름을 변경**



# 클래스(Class) vs 인스턴스(Instances)



class

color, size, is\_convertible, wheel\_size  
engine\_start(), engine\_stop(), accelerate(), break()



color="red"  
size = "medium"  
wheel\_size= 19



color="blue"  
size = "medium"  
wheel\_size= 19  
is\_convertible = "yes"

open\_roof(), close\_roof()

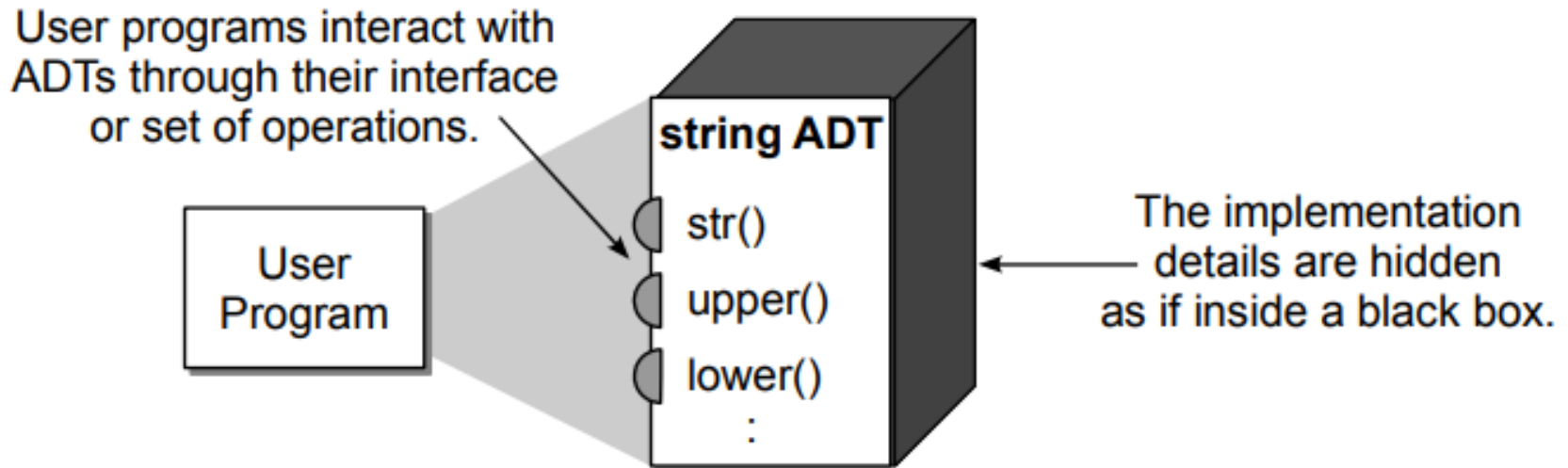


color="yellow"  
size = "medium"  
wheel\_size= 19

open\_roof(), close\_roof()



# 추상 자료형 (ADT: Abstract Data Type)



**Figure 1.2:** Separating the ADT definition from its implementation.



# 추상데이터 타입(ADT)

- 추상화란 사용자에게 중요한 정보는 강조되고 반면 중요하지 않은 구현 세부 사항은 제거하는 것
- 추상화(abstraction)-> 정보는닉기법(information hiding)-> 추상 자료형(ADT)
- 추상 자료형(ADT: Abstract Data Type)
  - 데이터 타입을 추상적(수학적)으로 정의한 것
  - 데이터나 연산이 무엇(what)인가를 정의함
  - 데이터나 연산을 어떻게(how) 구현할 것인지는 정의하지 않음



# Date의 추상화 타입

---

- Gregorian(양력) 날짜를 표현하는 ADT
  - Date(month, day, year): 년월일의 데이터 생성
  - day(): 일자 반환
  - month(): 월 반환
  - Year(): 년도 반환
  - monthName(): 영어로된 월 표현 반환, eg. March
  - dayOfWeek(): 요일 반환



```

2      # Gregorian(양력) 날짜를 표현하는 ADT
3      class Date :
4          ## 년월일의 데이터 생성
5          def __init__(self, month, day, year):
6              pass
7
8          # 년도 반환
9          def year(self):
10             pass
11         # 월 반환
12         def month(self):
13             pass
14
15         # 일자 반환
16         def day(self):
17             pass
18
19         # 영어로된 월 표현 반환, eg. March
20         def monthName(self):
21             pass
22
23         # 요일 반환
24         def dayOfWeek(self):
25             pass

```



# Date ADT의 구현

```
1  # Implements a proleptic Gregorian calendar date as a Julian day number.
2
3  class Date :
4      # Creates an object instance for the specified Gregorian date.
5      def __init__( self, month, day, year ) :
6          self._julianDay = 0
7          assert self._isValidGregorian( month, day, year ), \
8              "Invalid Gregorian date."
9
10         # The first line of the equation,  $T = (M - 14) / 12$ , has to be changed
11         # since Python's implementation of integer division is not the same
12         # as the mathematical definition.
13         tmp = 0
14         if month < 3 :
15             tmp = -1
16         self._julianDay = day - 32075 + \
17             (1461 * (year + 4800 + tmp) // 4) + \
18             (367 * (month - 2 - tmp * 12) // 12) - \
19             (3 * ((year + 4900 + tmp) // 100) // 4)
20
21         # Extracts the appropriate Gregorian date component.
22         def month( self ) :
23             return (self._toGregorian())[0] # returning M from (M, d, y)
24
25         def day( self ) :
26             return (self._toGregorian())[1] # returning D from (m, D, y)
27
28         def year( self ) :
29             return (self._toGregorian())[2] # returning Y from (m, d, Y)
30
31         # Returns day of the week as an int between 0 (Mon) and 6 (Sun).
32         def dayOfWeek( self ) :
33             month, day, year = self._toGregorian()
34             if month < 3 :
35                 month = month + 12
36                 year = year - 1
37             return ((13 * month + 3) // 5 + day + \
38                 year + year // 4 - year // 100 + year // 400) % 7
```



# 왜 파이썬인가?

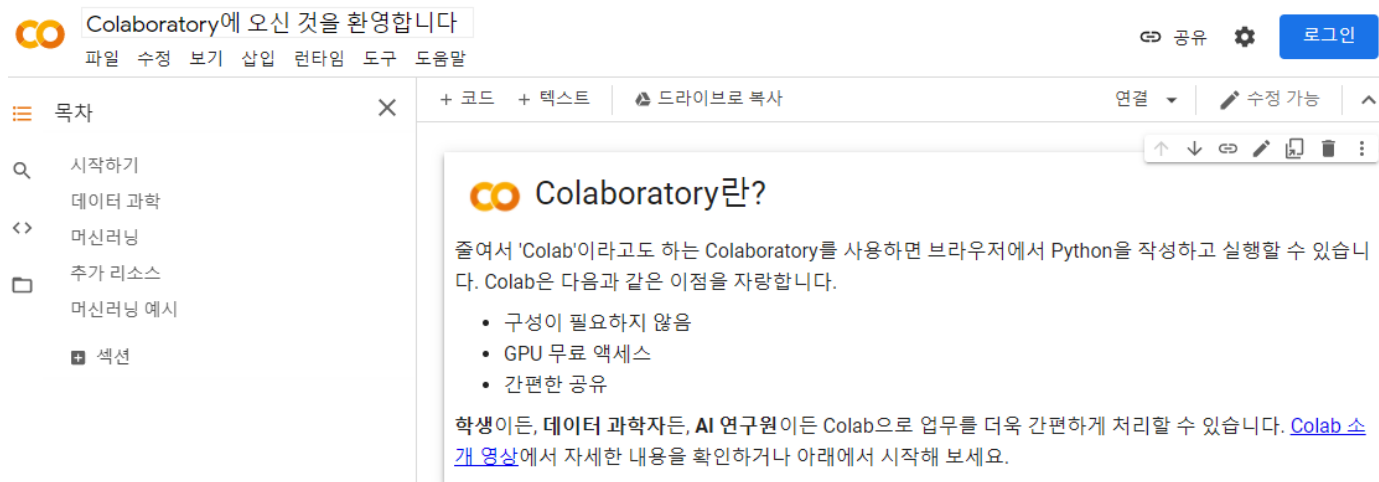


- Easy to Learn and Use
- Hundreds of Python Libraries and Frameworks
- Big data, Machine Learning and Cloud Computing



# Colab: 언제, 어디서든 프로그램을 빠르게 테스트

- Colab: 클라우드기반 무료 Jupyter 노트북
  - Python + 구글드라이브 + 구글클라우드+ 리눅스
- 가입절차
  - 구글 계정 필요(없다면 가입)
  - colab.research.google.com 이동



## ▼ 시작하기

지금 읽고 계신 문서는 정적 웹페이지가 아니라 코드를 작성하고 실행할 수 있는 대화형 환경인 Colab 메모장입니다.

예를 들어 다음을 값을 계산하여 변수로 저장하고 결과를 출력하는 간단한 Python 스크립트가 포함된 코드 셀입니다.





# 메뉴



Colaboratory에 오신 것을 환영합니다

파일 수정 보기 삽입 런타임 도구 도움말 변경사항을 저장할 수 없음

공유



목차



시작하기

데이터 과학



머신러닝



추가 리소스

머신러닝 예시

섹션

+ 코드 + 텍스트 드라이브로 복사

RAM 디스크

수정 가능

## 코드 or 텍스트 셀(cell) 삽입



### Colaboratory란?

줄여서 'Colab'이라고도 하는 Colaboratory를 사용하면 브라우저에서 Python을 작성하고 실행할 수 있습니다. Colab은 다음과 같은 이점을 자랑합니다.

- 구성이 필요하지 않음
- GPU 무료 액세스
- 간편한 공유

학생이든, 데이터 과학자든, AI 연구원이든 Colab으로 업무를 더욱 간편하게 처리할 수 있습니다. [Colab 소개 영상](#)에서 자세한 내용을 확인하거나 아래에서 시작해 보세요.

### 시작하기

지금 읽고 계신 문서는 정적 웹페이지가 아니라 코드를 작성하고 실행할 수 있는 대화형 환경인 Colab 메모장입니다.

예를 들어 다음은 값을 계산하여 변수로 저장하고 결과를 출력하는 간단한 드 셀입니다.

## 텍스트 셀

## 왼쪽패널



```
seconds_in_a_day = 24 * 60 * 60
seconds_in_a_day
```

86400

↑ ↓ 공유 설정 인출 삭제

## 코드 셀



# HW #1: 나만의 ADT 모델링

- 목표: 여러 아이템을 포함한 웹 콘텐츠를 ADT로 표현하기
- 내용: 22번 slide를 참고하여 캡처 이미지로부터 파이썬 기반의 ADT를 생성함
- 제출사항:
  - 모델링을 위한 캡처 이미지
  - ADT로 표현한 Colab내 텍스트 및 코드 생성
  - Colab 공유 기능을 통해 제출
- 평가조건 위반시 감점



# HW #1: 나만의 ADT 모델링

## ■ 평가

- 파일이름 변경: 데이터구조[분반] 학번 이름
  - 데이터구조[01] 2100000 홍길동
- 연속적인 아이템(데이터)를 포함하는 관련 이미지 1개 첨부
  - 5개 이상 다른 데이터 필드(Data field)를 포함
- 적절한 ADT 클래스 생성 및 인스턴스 출력
  - 변수를 추가하고, 32p를 참고하여 적절한 변수 유형을 표기할 것
  - print 함수를 사용하여 적어도 2개의 인스턴스를 출력
- [sejinc2021@gmail.com](mailto:sejinc2021@gmail.com) 과 공유(화면 우측 상단)
  - 잘못 공유시 미제출로 인정
- 제출 마감기간: 5월 31일까지



# HW #1: 변수 유형 참고

## 파이썬의 5가지 자료 구조, 변수 유형 (Python's 5 Data/Variable Types)

1 수 (Numbers)	2 문자열 (String)	3 리스트 (List)	4 튜플 (Tuple)	5 사전 (Dictionary)
정수(Int) 100	'I Love You'	['abc', 123]	('abc', 123)	{'name': 'R Friend', 'region': 'Seoul, Korea', 'phone': '02-123-3456'}
부동소수형(float) 12.345	<ul style="list-style-type: none"><li>- A sequence set of characters</li><li>- Enclosed by ' ', or ""</li></ul>	<ul style="list-style-type: none"><li>- 1 dimensional sequence of different data type objects</li><li>- <b>Can be updated</b></li><li>- Enclosed by []</li></ul>	<ul style="list-style-type: none"><li>- 1 dimensional sequence of different data type objects</li><li>- <b>Can NOT be updated</b></li><li>- Enclosed by ()</li></ul>	<ul style="list-style-type: none"><li>- Hash table type</li><li>- Associative array</li><li>- Key-value pairs</li><li>- Enclosed by {}</li></ul>
복소수(complex) 3.45j				

R분석과 프로그래밍 (<http://rfriend.tistory.com>)



제목: 웹툰 관리 프로그램을 만들고자 합니다.



**파이게임**  
배진수  
★★★★★ 9.75  
전체보기



**만엽록파**  
성불예경,홍실/미노  
★★★★☆ 8.63  
전체보기



**앵무살수**  
김성진  
★★★★★ 9.98  
전체보기



**착한건 돈어된다**  
유성연  
★★★★★ 9.74  
전체보기



**칼가는 소녀**  
오리  
★★★★★ 9.97  
전체보기



**유알무이 로맨스**  
두부  
★★★★★ 9.98  
전체보기

```
[2] class Webtoon:
    def __init__(self, title, authors, rating, link, img):
        self.title = title # 문자열
        self.authors = authors # 리스트
        self.rating = rating # 부동소수형(float)
        self.link = link # 문자열
        self.img = img # 문자열

w1 = Webtoon("파이게임", "배진수", 9.75, "http://abc.com", "img1.jpg")

print(w1.title)
print(w1.authors)
```

파이게임  
배진수