

강의	토픽	내용
2강	주석	한줄 주석 : # , 여러줄 주석 : ''' ''' , """ """
	기본출력 print	print('출력할 문자열') print(1004), print(2+3)
		print('문자열', '정수') # 여러 개 자료형 ,로 구분
		키워드 인수 : 출력 자료의 구분 문자 지정 print(value, ..., sep=' ', end='\n')
3강	변수	변수 = 자료, 변수 = 수식, 변수 = 함수 수행 결과 x = 100, y=200 x, y = 100, 200 #여러 변수 동시 선언
	기본 입력 input	변수 = input('프롬프트_메시지')
	자료 변환 함수	변수 = int('숫자로구성된문자열') # 인수를 정수로 변환 변수 = float('숫자로구성된문자열') # 인수를 실수로 변환
	연산자	산술: +, -, *, //, /, %, ** 비교 : < > <= >= == != 논리 : and, or, not
4강	문자열 표현	두 종류의 따옴표 사용 가능 '...' 또는 "..." 이스케이프 문자 - W, W", WWW, Wn, Wt
	인덱싱	0 에서 시작, -1 :역순가능 msg = "안녕하세요", msg[0]
	슬라이싱	시작 부터 (끝-1) 까지, msg[0:2] => 안녕
	연산 +, *	'문자열_리터럴1' '문자열_리터럴2' 또는 문자열1 + 문자열2 # 두 문자열을 연결 문자열 * 정수
	f 문자열	f'... {표현식} ...' # 표현식의 값을 갖는 문자열을 결과값으로 가짐 문자열: s, 정수 : d, 실수 : f, 지수 : e, 이진 : b 왼쪽 정렬 : <, 가운데 정렬 : ^, 오른쪽 정렬 : >, 0채움 : 0 print(f'{num:5d}'), print(f'{Pi:.2f}')
	문자열함수	len() - 길이, split() - 분리, count() - 갯수, find() - 검색
5강	if/if ~ else/ if~elif~else	<div> <pre>if n1 > n2 : print(f'{n1}') print('크다')</pre> </div> <div> <pre>if n1 > n2 : print(f'{n1}') else : print(f'{n2}')</pre> </div> <div> <pre>if 조건문 1: 조건 1이 만족하 elif 조건문 2: 조건 2이 만족하 ... else: 모든 조건이 만</pre> </div>
6강	while/ for	<div> while 조건평가식: 제어를 받는 문장 일반적인 문장 while pwd != '#12*': pwd = input('passwd?') </div> <div> for 변수 in 범위표현: 제어를 받는 문장 일반적인 문장 for i in [1,2,3,4,5] print('hello') </div> <div> for 변수 in range(값): 제어를 받는 문장 일반적인 문장 for i in range(5): print('hello') </div>

7강	중첩 & 무한루프	<div><pre>for m in range(1,5): #... for n in range(1,4): #...</pre></div> <div>외부반복</div> <div>내부반복</div>	<pre>while True : if 조건 : break if 조건 : continue</pre>																								
	보조제어문	break: 반복문에서 빠져나감 continue: 다음 반복시작																									
9강	함수	<div><pre>def 함수이름(): 수행할 문장1 수행할 문장2</pre></div> <div><pre>def show_message() : print('hello world!') print('good job!')</pre></div> <div># 주 프로그램부 <pre>print('시작') show_message()</pre></div>																									
	매개변수 vs 리턴값	<div><pre>def get_area(radius) : area = 3.14*radius**2 return area</pre></div> <div>[반환값 있음]</div> <div>return 문장은 함수를 종료시키고 결과를 반환한다.</div> <div><pre># 함수 호출 result = get_area(3) 인수</pre></div> <div>결과대입</div>																									
10강	지역변수/전역변수	지역변수 : 함수정의 같은 특정 블록안에서 선언된 변수 전역 변수 : 어떤 블록에도 속하지 않게 선언된 변수 # 지역 변수에서 전역변수를 사용하여 값을 할당시 : global																									
	모듈 구성	1. 연관된 함수를 별도의 파일로 만들 : calc_area.py 2. import 사용하여 지정한 모듈 가져오기 : import calc_area <pre>import 모듈이름 from 모듈이름 import : import 모듈이름 as 모듈_별명</pre>																									
	표준모듈	random random.randint(1, 6)																									
11강 / 12강		<table><thead><tr><th></th><th>리스트 / 인덱스</th><th>딕셔너리 / 키</th></tr></thead><tbody><tr><td>① 생성</td><td>• nums = [11, 22, 33, 44, 55]</td><td>• dic = {'a':'apple', 'g':grape}</td></tr><tr><td>② 요소접근</td><td>• nums[1], nums[1:3]</td><td>• dic['a']</td></tr><tr><td>③ 추가</td><td>• nums.append('66')</td><td>• dic['k'] = 'kiwi'</td></tr><tr><td>④ 수정</td><td>• nums[1]=99, nums[1:3]=[2,3]</td><td>• dic['a'] = 'avocado'</td></tr><tr><td>⑤ 삭제</td><td>• del nums[1], nums.clear()</td><td>• del dic['g'], dic.clear()</td></tr><tr><td>⑥ 유효성검사</td><td>• 11 in nums</td><td>• 'a' in dic</td></tr><tr><td>⑦ 반복(for)</td><td>• for item in nums: print(item) • for i in range(len(nums)): print(nums[i])</td><td>• for item in dic.items(): print(item) #키:값 • for key in dic.keys(): print(key, dic[key])</td></tr></tbody></table>		리스트 / 인덱스	딕셔너리 / 키	① 생성	• nums = [11, 22, 33, 44, 55]	• dic = {'a':'apple', 'g':grape}	② 요소접근	• nums[1], nums[1:3]	• dic['a']	③ 추가	• nums.append('66')	• dic['k'] = 'kiwi'	④ 수정	• nums[1]=99, nums[1:3]=[2,3]	• dic['a'] = 'avocado'	⑤ 삭제	• del nums[1], nums.clear()	• del dic['g'], dic.clear()	⑥ 유효성검사	• 11 in nums	• 'a' in dic	⑦ 반복(for)	• for item in nums: print(item) • for i in range(len(nums)): print(nums[i])	• for item in dic.items(): print(item) #키:값 • for key in dic.keys(): print(key, dic[key])	
		리스트 / 인덱스	딕셔너리 / 키																								
	① 생성	• nums = [11, 22, 33, 44, 55]	• dic = {'a':'apple', 'g':grape}																								
	② 요소접근	• nums[1], nums[1:3]	• dic['a']																								
	③ 추가	• nums.append('66')	• dic['k'] = 'kiwi'																								
	④ 수정	• nums[1]=99, nums[1:3]=[2,3]	• dic['a'] = 'avocado'																								
	⑤ 삭제	• del nums[1], nums.clear()	• del dic['g'], dic.clear()																								
	⑥ 유효성검사	• 11 in nums	• 'a' in dic																								
⑦ 반복(for)	• for item in nums: print(item) • for i in range(len(nums)): print(nums[i])	• for item in dic.items(): print(item) #키:값 • for key in dic.keys(): print(key, dic[key])																									
파일 열기	파일객체이름 = open('대상파일이름', mode='열기모드')																										
파일 닫기	파일객체이름.close()																										
with로 파일 안전하게 닫기	with open('파일이름', '열기모드') as 파일객체이름: # 파일객체를 가지고 파일 입출력 처리																										
파일 쓰기	outfile.write(), outfile.readlines()																										
파일 읽기	infile.read(), infile.readlines(), infile.readline()																										
텍스트에서 한줄 읽어오기	for line in file:← lines.append(line.strip('\n'))																										
13강	파일 열기	파일객체이름 = open('대상파일이름', mode='열기모드')																									
	파일 닫기	파일객체이름.close()																									
	with로 파일 안전하게 닫기	with open('파일이름', '열기모드') as 파일객체이름: # 파일객체를 가지고 파일 입출력 처리																									
	파일 쓰기	outfile.write(), outfile.readlines()																									
	파일 읽기	infile.read(), infile.readlines(), infile.readline()																									
	텍스트에서 한줄 읽어오기	for line in file:← lines.append(line.strip('\n'))																									