|  |  |
| --- | --- |
| 교육 제목 | AI 프로젝트 기반 S/W 전문가 양성과정 |
| 교육 일시 | 21.0.29 |
| 교육 장소 | 대면(강의실) |
| **교육 내용** | |
| 오전 | 복습  print() : 콘솔에 문자열 출력  input() : 문자열 입력 받음  len() : 문자열의 길이 반환  숫자와 문자열의 다양한 기능  format() : 숫자 데이터를 문자열 형식으로 중괄호에 값을 대입  #index error : 괄호기호 개수가 format보다 많으면 발생  반대로 format이 많으면 일부만 변환  # 정수 자리 수 지정하기  print("{:5d}".format(100))  ->출력 범위를 5자리로 정의. C언어로 보면 %5d  #부호 강제로 표기하기  print("{:+d}".format(16)) :+16  print("{:+d}".format(-16)) :-16  #부호자리 비워 두기. 양수는 +가 출력 되지 x  print("{: d}".format(16)) : 16  print("{: d}".format(-16)) :+16  # 0번째 자리에 기호 박기  print("{:=+5d}".format(52)) :+0052  print("{:=+5d}".format(-52)) :-0052  소수점 출력  print("{:f}".format(3.1415))  소수점 미만 자리 수 카운트 지정, 정수.소수  print("{:15.3f}".format(3.1415))  의미 없는 소수점 제거  print("{:g}".format(3.1415))  문자열 관련 함수  string.함수()    upper->모든 알파벳을 대문자로 전환  lower->모든 알파벳을 소문자로 전환  strip-> 양쪽 공백 제거  lstrip->왼쪽 공백 제거  rstrip->오른쪽 공백 제거  #조합하기. 파이썬은 객체지향이라 연달아 붙이기가 가능하다  Ex)소문자로 바꾸고 양쪽 공백 제거  print("#",string\_data.lower().strip(),"#")  문자열 구성 파악 하기. is함수명(), 반환은 bool type data  isalnum() : 문자열이 알파벳 또는 숫자로만 구성되어 있는지  isalpha() : 문자열이 알파벳으로만 이루어져 있는지  isdentifier() : 문자열이 식별자로 사용 할 수 있는지  isdecimal(): 문자열이 정수 형태인지  isdigit() : 문자열이 숫자로 인식 될 수 있는건지  isspace() : 문자열이 공백으로만 구성 되어 있는지  islower() : 문자열이 소문자로만 구성 되어 있는지  isupper() : 문자열이 대문자로만 구성 되어 있는지  문자열 찾기  find() : 왼쪽에서 부터 찾기. 리턴 값은 배열 인덱스  rfind() : 오른쪽에서 부터 찾기. 리턴 값은 배열 인덱스  split-> 문자열 자르기, list 형식으로 반환  아무 것도 안 적으면 공백 단위로 분해.  Bool 연산  비교 연산자  < , = , > , >= , == , <=, !=  논리 연산자  AND  OR  NOT  If조건문  파이썬에서 if 조건문은 들여쓰기를 중요시함. 주의 요망  if <조건문>:  <처리할 내용>  elif <조건문>:  <처리할 내용>  ...  elif <조건문>:  <처리할 내용>  else:  <처리할 내용> |
| 오후 | 날짜/시간 활용하기  # 모듈을 사용하기 위해 추가  import datrtime > datetime  #현재 시간을 변수에 저장  get\_date = datrtime.datrtime.now()  #변수 사용  get\_date.<시간형식>  list 자료형  문자열 처럼 0부터 시작하며 +,x 연산이 가능하다  append() -> list 마지막에 요소 추가  리스트.append(요소)  insert() -> 리스트 특정 부위에 요소 추가  리스트.insert(위치,요소) |