|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 6차시 | 1 | 번 |  | 연습: ☑ | 과제 : □ | 평가 : □ |
| 문제1) 함수를 사용하면 좋은 점이 아닌 것은?  (1) 코드 재사용이 쉬워짐  **(2) 흐름을 단순화 시켜 함수를 아무리 많이 호출해도 안정적으로 운영됨**  (3) 함수 단위로 검수할 수 있고, 전체적으로 안정적인 프로그램을 작성할 수 있게 해줌  (4) 작업을 작은 단위로 나누어서 함수로 처리함으로써 문제의 크기를 줄여서 해결하는 것이 가능    문제2) 다음의 add 함수를 호출하는 결과를 예측해보고, 코딩하여 결과를 확인해보세요.    def add(num1, num2, num3):  return num1 + num2 + num3    add(2, 3, 5)  **10**  add(2, num3 = 5, num2 = 3)  **10**  add(2, num3 = "5", num2 = "3")  **Traceback (most recent call last):**  **File "<stdin>", line 1, in <module>**  **File "<stdin>", line 2, in add**  **TypeError: unsupported operand type(s) for +: 'int' and 'str'**  add(2, num2 = 3, 5)  **File "<stdin>", line 1**  **add(2, num2 = 3, 5)**  **^**  **SyntaxError: positional argument follows keyword argument**  문제3) print(2, 3, sep = ",") 명령문의 출력을 적으시오.  **정답3) 2,3**  문제4) 다음 코드는 오류가 발생하는지 설명하시오.    >>> num = 3  >>> def printNum3():  ... print(num)  ... num = 5  ... print(num)  ...  >>> printNum3() | | | | | | |

|  |
| --- |
| **정답4)오류 발생**  **>>> num = 3 # (1)**  **>>> def printNum3():**  **... print(num) # (2)**  **... num = 5 # (3)**  **... print(num)...**  **>>> printNum3()**  **Traceback (most recent call last):**  **File "<stdin>", line 1, in module**  **File "<stdin>", line 2, in printNum3**  **UnboundLocalError: local variable 'num' referenced before assignment**  **오류 발생 이유:**  **파이썬에서 함수 내에 변수가 만들어지면 그 변수의 유효 영역(scope)은 함수 전체가**  **된다.**  **num변수가(1)에서 전역 변수로 만들어졌다.**  **printNum3()의(2)코드에서 사용되는num변수는(1)에서 만들어진 변수인 것처럼**  **보이지만,실제로는(3)에서 만들어진 변수이다. -->함수 내에 변수가 만들어지면 그**  **변수의 유효 영역이 함수 전체가 되기 때문이다.**  문제5) 1부터 100까지의 합계를 계산하는 재귀함수를 작성한다.  **정답5)**  **def sum1ToN(n):**  **if n != 1:**  **return n + sum1ToN(n - 1)**  **else:**  **return 1**  **print(sum1ToN(100))**  문제6) 이슬점을 구하는 함수를 구현하고, 사용자로부터 입력 받은 습도와 온도로 이슬점을 구하는 프로그램을 작성하시오. 이슬점을 구하는 함수는 습도와 온도를 인자로 전달받고 이슬점을 반환한다.    이슬점을 구하는 수식:  import math  d1 = math.log(humid / 100)  d2 = (17.62 \* temperature) / (243.12 + temperature)  이슬점 = (243.12 \* (d1 + d2)) / (17.62 - (d1 + d2))  **정답6)**  **import math**  **def getDewPoint(temperature, humid):**  **d1 = math.log(humid / 100)**  **d2 = (17.62 \* temperature) / (243.12 + temperature)**  **return (243.12 \* (d1 + d2)) / (17.62 - (d1 + d2))**  문제7) 날짜를 출력하는 함수를 구현한다. 함수는 다음 형태로 동작한다.  >>> PrintDate(2022, 8, 1)  Year: 2022  Month: August  Day: 1    구현된 함수를 이용해서 2022년 9월 20일과 2023년 3월 3일을 출력하는 프로그램을 작성한다.  **정답7)**  **def PrintDate(year, month, day):**  **print(f"Year: {year}")**  **if month == 1:**  **m = "January"**  **elif month == 2:**  **m = "February"**  **elif month == 3:**  **m = "March"**  **elif month == 4:**  **m = "April"**  **elif month == 5:**  **m = "May"**  **elif month == 6:**  **m = "June"**  **elif month == 7:**  **m = "July"**  **elif month == 8:**  **m = "August"**  **elif month == 9:**  **m = "September"**  **elif month == 10:**  **m = "October"**  **elif month == 11:**  **m = "November"**  **elif month == 12:**  **m = "December"**  **print(f"Month: {m}")**  **print(f"Day: {day}")**  **PrintDate(2022, 9, 20)**  **PrintDate(2023, 3, 3)**  문제8) 주어진 문자열에서 단어의 개수를 세는 함수를 작성하세요. 단어는 공백으로 구분됩니다.  **정답8)**  **def count\_words(s):**  **return len(s.split())**  **#예제 출력**  **print(count\_words("안녕하세요 여러분")) # 2**  **print(count\_words("이것은 파이썬 연습문제입니다")) # 3**  문제9) 함수의 가변 인자에 대해 옳지 않은 설명을 고르시오.  (1) 가변 인자는 함수 호출 시 인자의 개수를 동적으로 처리할 수 있게 해줍니다.  (2) 파이썬에서는 \*args 형태로 가변 인자를 정의하며, 튜플 형태로 처리됩니다.  (3) 가변 인자를 사용하면 함수 정의 시 모든 인자를 필수적으로 선언할 필요가 없습니다.  **(4) 가변 인자를 사용할 때에는 튜플 형태로 인자를 전달하는 것이 아니라, 각 인자들이 함수에 리스트 형태로 전달됩니다.** |

|  |
| --- |
| 문제10) 세 개 단어로 구성된 문자열을 첫번째 인자로, 1부터 3까지의 정수값을 두번째 인자로 전달 받는 함수 “getWord”를 구현한다. 함수는 두번째 인자 위치에 해당되는 단어를 문자열로 반환한다.    >>> print(getWord("A beautiful day", 1))  A  >>> print(getWord("A beautiful day", 3))  day  **정답10)**  **# s는 세 개 단어로 구성되어 있다고 가정**  **def getWord(s, pos):**  **news = s.strip()**  **if pos == 1:**  **idx = news.find(' ')**  **return news[:idx]**  **elif pos == 2:**  **idx = news.find(' ') #첫 번째 공백문자 찾기**  **news = news[idx + 1:].strip() #첫 번째 단어를 제외한 나머지 문자열**  **idx = news.find(' ') #두 번째 공백문자 찾기**  **return news[:idx]**  **elif pos == 3:**  **idx = news.rfind(' ')**  **return news[idx + 1:]**  **print(getWord("A beautiful day", 1))**  **print(getWord("A beautiful day", 3))**  문제11) 정수 한 개를 함수의 매개변수를 통해 입력받고, 윤년인지 확인해서 True 또는 False를 반환하는 함수를 구현하고 이를 검수하는 프로그램을 작성하시오.  <윤년의 조건>  - 연도가 4로 나누어지면 윤년  - 연도가 4로 나누어지면서 100으로 나누어지면 윤년 아님  - 연도가 400으로 나누어지면 윤년  **정답11)**  **def isLeapYear(year):**  **return (year % 400 == 0) or (year % 4 == 0 and year % 100 != 0)**  **print(isLeapYear(1234))**  **print(isLeapYear(1900))**  **print(isLeapYear(2000))**  **print(isLeapYear(2204))**  문제12) 두 개 이상의 단어가 있는 문자열을 입력으로 전달받고, 두 번째 단어만 추출해서 반환하는 함수를 구현하고 이를 검수하는 프로그램을 작성하시오.  <요구 사항>  - 함수에 전달되는 문자열은 반드시 두 개 이상의 단어로 구성되어 있다고 가정  - 단어는 공백 문자, 웹 문자, 줄바꿈 문자 중 한 개로 분리됨(중복되지 않음)  - 문자열의 양끝에 공백 문자가 있을 수 있음(제거 후 단어를 추출)  **정답12)**  **def getSecondWord(s):**  **s = s.strip()**  **pos1 = s.find(' ')**  **pos2 = s.find('\t')**  **pos3 = s.find('\n')**  **# -1보다 큰 값을 반환함**  **p1 = max(pos1, pos2, pos3) + 1**  **s2 = s[p1:]**  **pos1 = s2.find(' ')**  **pos2 = s2.find('\t')**  **pos3 = s2.find('\n')**  **p2 = max(pos1, pos2, pos3)**  **if p2 == -1:**  **p2 = len(s)**  **else:**  **p2 += p1**  **return s[p1:p2]**  **print(getSecondWord("what a beautiful day"))**  **print(getSecondWord("beautiful day"))**  문제13) 세 명의 성적을 입력으로 전달받고, 화면에 순서대로 출력하는 함수를 작성하고, 이 함수를 호출하는 코드를 구현하시오.  - 일반적으로 성적은 내림차순으로 출력하지만, 가끔씩 오름차순으로 출력하는 경우도 있음. 매개 변수를 이용해서 결정할 수 있도록 할 것  <요구사항>  - 매개변수의 기본값을 내림차순으로 지정  **정답13)**  **def CompareScores(score1, score2, score3, order = True):**  **if order:**  **if score1 >= score2 and score1 >= score3:**  **if score2 >= score3:**  **print(score1, score2, score3)**  **else:**  **print(score1, score3, score2)**  **elif score2 >= score1 and score2 >= score3:**  **if score1 >= score3:**  **print(score2, score1, score3)**  **else:**  **print(score2, score3, score1)**  **else:**  **if score1 >= score2:**  **print(score3, score1, score2)**  **else:**  **rint(score3, score2, score1)**  **else: #오름차순**  **if score1 < score2 and score1 < score3:**  **if score2 < score3:**  **print(score1, score2, score3)**  **else:**  **print(score1, score3, score2)**  **elif score2 < score1 and score2 < score3:**  **if score1 < score3:**  **print(score2, score1, score3)**  **else :**  **print(score2, score3, score1)**  **else:**  **if score1 < score2:**  **print(score3, score1, score2)**  **else:**  **print(score3, score2, score1)**  **print("내림차순")**  **CompareScores(85, 80, 90)**  **print("오름차순")**  **CompareScores(85, 80, 90, False)**  문제14) 피보나치 수열을 작성하는 프로그램을 작성하시오.  - 피보나치 수열: 1 1 2 3 5 8 13 21 34 55 89 ...  - 1항과 2항은 1  - 3항 이후부터의 n항은 (n-1)항 + (n-2)항  \* f(n) = f(n-1) + f(n-2)  **정답14)**  **def fibonacci(n):**  **if n <= 0:**  **return 0**  **elif n == 1:**  **return 1**  **else:**  **return fibonaccie(n-1) + fibonacci(n-2)**  **#피보나치 수열의 첫10개 항을 출력**  **for i in range(10):**  **print(fibonacci(i))** |

|  |
| --- |
| 문제15) 최대공약수(Great Common Denominator) 구하기  <알고리즘>  - m = n이면 m 또는 n 반환  - m > n이면 m –n과 n의 최대공약수 반환  - m < n이면 m과 n –m의 최대공약수 반환  **정답15)**  **def gcd(m, n):**  **if m == n:**  **return m #재귀호출 탈출 조건**  **elif m > n:**  **return gcd(m - n, n)**  **else: # m < n**  **return gcd(m, n - m)**  **print(gcd(12, 4)) #4**  **print(gcd(12, 18)) #6**  문제16) 재귀함수를 사용하여 주어진 문자열을 뒤집은 결과를 반환하는 함수를 구현하세요  **정답16)**  **def reverse\_string(s):**  **if len(s) <= 1:**  **return s**  **else:**  **return reverse\_string(s[1:]) + s[0]**  **#예제 출력**  **print(reverse\_string("hello")) # "hello"를 뒤집은 결과는"olleh"**  문제17) 아래의 코드를 실행했을 때 어떤 결과가 나오는지 예측해 보세요.  def hello(\*names):  for each in names: #names 내의 모든 요소들을 순서대로 참조하는 순환문  print('안녕, {}!'.format(each))    hello('민정')  hello('David','Veronica','Paul')  hello('방탄소년단','블랙핑크')  **정답17)**  **안녕,민정!**  **안녕, David!**  **안녕, Veronica!**  **안녕, Paul!**  **안녕,방탄소년단!**  **안녕,블랙핑크!**  문제18) 재귀함수를 사용해서 팩토리얼을 계산하는 함수를 구현하세요.  **정답18)**  **def factorial(n):**  **if n == 0:**  **return 1**  **else:**  **return n \* factorial(n - 1)**  **#예제 출력**  **print(factorial(5)) # 5! = 120** |