# 13차 연습문항

## **파이썬 기본문항: 모듈**

|  |
| --- |
| 수강생 공지 사항   * 제출은 프로그래머스를 통해 해주시기 바랍니다([링크](https://campus.programmers.co.kr/app/courses/24550/curriculum)). * 파일명은 아래와 같은 형식으로 제출해주세요   + 교육생번호\_이름\_교과목\_문항\_N차시\_강의명.pdf   ex) DR-11111\_홍길동\_파이썬\_연습문항\_1차시\_환경및기본.pdf  ex) DR-11111\_홍길동\_파이썬\_추가문항\_1차시\_환경및기본.pdf   * 답은 "write your answer"에 적어주세요. 다만 코딩 문제의 경우 output까지 답에 포함시켜 주세요.   ex)     * 답은 캡쳐를 하셔도 되고 텍스트로 넣으셔도 됩니다. * 마감 기한은 문제가 나간 주 **일요일 23:59까지**입니다. |

### 파이썬에서 모듈의 크기에 대한 설명 중 올바른 것은 무엇입니까?

(1) 모듈은 항상 작은 크기로 제한되어야 한다.

(2) 모듈의 크기는 소스 코드 파일의 크기와 동일하다.

(3) 모듈의 크기는 포함된 기능의 수에 따라 결정된다.

(4) 모듈의 크기는 특정하지 않으며, 기능의 논리적 그룹에 따라 다를 수 있다.

**답**

|  |
| --- |
| (4) 모듈의 크기는 특정하지 않으며, 기능의 논리적 그룹에 따라 다를 수 있다. |

### 다음 코드에서 Rectangle.Rectangle로 이름이 겹쳐지고 있다. 앞과 뒤의 Rectangle이 각각 무엇인지 고르시오.

|  |
| --- |
| Example Code |
| import Rectangle  r = Rectangle.Rectangle(10, 20, 20, 10)  print(r.calcArea()) |

{1) 클래스-모듈

(2) 모듈-클래스

(3) 클래스-객체

(4) 모듈-객체

**답**

|  |
| --- |
| (2) |

### 모듈을 사용하는 이유가 아닌 것은 모두 고르시오.

(1) 코드 재사용성을 높이기 위해

(2) 코드의 가독성을 개선하기 위해

(3) 관련 객체들의 결합성을 증가시키기 위해

(4) 대규모 프로젝트의 관리를 용이하게 하기 위해

(5) 코드의 실행 속도를 빠르게 하기 위해

(6) 특정 기능이나 로직을 독립적으로 테스트하기 위해

**답**

|  |
| --- |
| (3) (5) |

### 다른 사람들이 만들어둔 패키지(모듈들)의 파이썬 설치 관리 도구를 무엇이라고 하는가?

(1) pip

(2) virtualenv

(3) conda

(4) npm

**답**

|  |
| --- |
| (1) |

### 다음 중 파이썬의 time 모듈에서 "epoch"에 대한 설명으로 옳은 것은 무엇입니까?

(1) Epoch는 모든 컴퓨터 시스템에서 2000년 1월 1일 00:00:00을 기준으로 하는 시간의 시작점을 나타낸다.

(2) Epoch는 모든 컴퓨터 시스템에서 동일하게 1970년 1월 1일 00:00:00 UTC를 기준으로 하는 시간의 시작점을 나타낸다.

(3) Epoch는 컴퓨터 시스템에 따라 다를 수 있지만, 대부분의 유닉스 기반 시스템 에서는 1970년 1월 1일 00:00:00 UTC를 기준으로 한다.

(4) Epoch는 시스템의 현재 시간을 기준으로 설정된다.

**답**

|  |
| --- |
| (3) |

### glob 모듈을 이용하면 유닉스나 윈도우의 커맨드 라인에서 사용하는 와일드카드를 사용할 수 있다. 다음을 와일드카드를 사용해 작성하세요.

(1) "bc.txt"로 끝나는 3글자 항목 ex) abc.txt, qbc.txt

(2) 확장자 ".txt"를 가진 모든 파일

(3) 모든 항목

**답**

|  |
| --- |
| (1) ?bc.txt  (2) \*.txt  (3) 모든 파일을 나타내는 와일드 카드가 윈도우와 유닉스에서 좀 다르게 쓰인다. 윈도우에서는 \*.\*  유닉스 계열에서는 \* |

### 다음 중 명령형 인자(command-line argument)에 대한 옳은 설명은 무엇입니까?

(1) 명령행 인자는 프로그램이 실행될 때 외부에서 추가적인 정보나 데이터를 전달하는 방법이다.

(2) 명령행 인자는 오직 숫자형 데이터만을 전달할 수 있다.

(3) 명령행 인자는 반드시 프로그램의 코드 내에서 하드코딩되어 있어야 한다.

(4) 명령행 인자는 프로그램이 실행될 때 자동으로 생성되어 사용자에게 보이지 않는다.

**답**

|  |
| --- |
| (1) |

### datetime 모듈을 사용하여 다음을 출력하는 코드를 작성하세요.

(1) 오늘의 년도, 월, 일

(2) 현재 일자를 형식에 맞추어 출력 (예: 2022년 08월 18일 요일: Thursday)

(3) 현재 시각을 형식에 맞추어 출력 (예: 12시 35분 25초)

**답**

|  |
| --- |
| import datetime    # (1)  d = datetime.date.today()  print(d)    # (2)  print(d.strftime("%Y년 %m월 %d일 요일: %A"))    # (3)  currentTime = datetime.datetime.today()  currentTime.strftime("%H시 %M분 %S초") |

### 정해진 기준(1970년 1월 1일 00:00:00 UTC)이후 현재까지 몇 초가 흘렀는지 출력하는 코드를 작성하세요.

**답**

|  |
| --- |
| import time  print(time.time()) |

### 현재 작업 디렉토리를 출력하는 코드를 적으세요.

**답**

|  |
| --- |
| import os  print(os.getcwd()) |