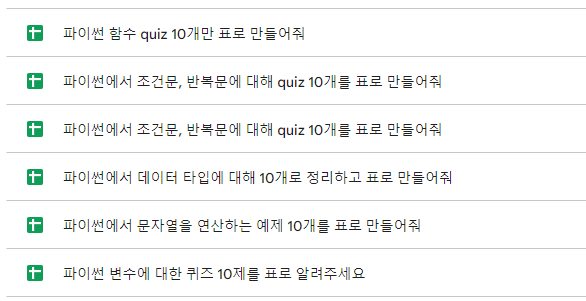
**1. Bard로 뽑아보는 파이썬 Quiz**

****

bard에게 각 분야별로 10개의 Quiz를 요청한 결과를 정리함. 잘못된결과와 중복된 결과를 제거하고 깔끔하게 표로 정리함.

1. 파이썬 변수 10제

| 문제 | 정답 |
| --- | --- |
| 파이썬에서 변수를 만드는 방법은? | 변수명 = 값 |
| 파이썬에서 변수의 이름을 지을 때는 어떤 규칙이 있는가? | 변수의 이름은 영문자, 숫자, 밑줄로만 이루어질 수 있으며, 첫 글자는 영문자여야 한다. |
| 파이썬에서 변수의 범위는? | 변수는 변수가 선언된 블록 내에서만 유효하다. |
| 파이썬에서 변수의 값을 바꾸는 방법은? | 변수명 = 새로운 값 |
| 파이썬에서 변수의 값을 출력하는 방법은? | print(변수명) |
| 파이썬에서 변수의 유효성을 검사하는 방법은? | type(변수명) |
| 파이썬에서 변수를 삭제하는 방법은? | del 변수명 |
| 파이썬에서 변수의 리스트를 얻는 방법은? | globals() |
| 파이썬에서 변수의 값을 키워드 인자로 전달하는 방법은? | 함수명(\*\*변수명) |

2. 파이썬 변수 데이터 타입 10제

| 데이터 타입 | 설명 |
| --- | --- |
| None | 값이 없음을 나타내는 데이터 타입입니다. |
| Boolean | 참 또는 거짓을 나타내는 데이터 타입입니다. |
| Integer | 정수를 나타내는 데이터 타입입니다. |
| Float | 실수를 나타내는 데이터 타입입니다. |
| Complex | 복소수를 나타내는 데이터 타입입니다. |
| String | 문자열을 나타내는 데이터 타입입니다. |
| List | 리스트는 순서가 있는 값들의 집합을 나타내는 데이터 타입입니다. |
| Tuple | 튜플은 순서가 있는 값들의 고정된 집합을 나타내는 데이터 타입입니다. |
| Set | 세트는 순서가 없는 값들의 집합을 나타내는 데이터 타입입니다. |
| Dictionary | 사전은 키와 값의 쌍으로 이루어진 데이터 타입입니다. |

3. 파이썬 연산자 10제

| 문제 | 답 |
| --- | --- |
| 파이썬에서 산술 연산자는 무엇입니까? | +, -, \*, /, %, \*\* |
| 파이썬에서 관계 연산자는 무엇입니까? |  |
| 파이썬에서 논리 연산자는 무엇입니까? | and, or, not |
| 파이썬에서 조건 연산자는 무엇입니까? | if, else, elif |
| 파이썬에서 증감 연산자는 무엇입니까? | ++, -- |
| 파이썬에서 비트 연산자는 무엇입니까? | &, ` |
| 파이썬에서 시퀀스 연산자는 무엇입니까? | +, \*, in, not in |
| 파이썬에서 멤버십 연산자는 무엇입니까? | in, not in |
| 파이썬에서 인덱싱 연산자는 무엇입니까? | [] |

4.. 파이썬 문자열 연산 10제

| 연산 | 설명 |
| --- | --- |
| len() | 문자열의 길이를 반환합니다. |
| str1 + str2 | 두 문자열을 연결합니다. |
| str1 \* n | 문자열을 n번 반복합니다. |
| str1[start:end] | 문자열의 일부분을 반환합니다. |
| str1.find(str2) | 문자열 str2가 문자열 str1에 처음 나타나는 위치를 반환합니다. |
| str1.replace(str1, str2) | 문자열 str1에서 문자열 str2를 문자열 str3로 바꿉니다. |
| str1.upper() | 문자열을 대문자로 변환합니다. |
| str1.lower() | 문자열을 소문자로 변환합니다. |
| str1.split() | 문자열을 공백 문자로 구분하여 리스트로 반환합니다. |

5. 파이썬 조건문, 반복문 10제

| 문제 | 정답 |
| --- | --- |
| 다음 중 조건문은? | if, elif, else |
| 다음 중 반복문은? | while, for |
| 다음은 조건문의 예시입니다. | if x > 10:  print("x is greater than 10") |
| 다음은 반복문의 예시입니다. | for i in range(10):  print(i) |
| 다음은 조건문과 반복문을 함께 사용하는 예시입니다. | while x > 0:  x -= 1 print(x) |
| 다음은 조건문과 반복문을 함께 사용하여 1부터 10까지의 합을 구하는 코드입니다. | sum = 0  for i in range(1, 11):  sum += i   print(sum) |
| 다음은 조건문과 반복문을 함께 사용하여 1부터 10까지의 짝수의 합을 구하는 코드입니다. | sum = 0  for i in range(2, 11, 2):  sum += i  print(sum) |
| 다음은 조건문과 반복문을 함께 사용하여 1부터 10까지의 홀수의 합을 구하는 코드입니다. | sum = 0  for i in range(1, 11, 2):  sum += i  print(sum) |
| 다음은 조건문과 반복문을 함께 사용하여 1부터 10까지의 3의 배수의 합을 구하는 코드입니다. | sum = 0  for i in range(3, 11, 3):  sum += i  print(sum) |
| 다음은 조건문과 반복문을 함께 사용하여 1부터 10까지의 5의 배수의 합을 구하는 코드입니다. | sum = 0  for i in range(5, 11, 5):  sum += i  print(sum) |

6. 파이썬 함수 5제

| 문제 | 정답 |
| --- | --- |
| 파이썬 함수는 어떻게 정의합니까? | def 함수명(매개변수):   코드블록 |
| 함수의 매개변수는 어떻게 전달합니까? | 함수명(매개변수값) |
| 함수는 어떻게 호출합니까? | 함수명() |
| 함수는 어떻게 리턴합니까? | return 리턴값 |
| 함수는 어떻게 가변 키워드 인자를 전달합니까? | 함수명(\*\*인자값) |

7. 파이썬에서 배열 10제

| 문제 | 답 |
| --- | --- |
| 다음 중 파이썬에서 배열을 생성하는 방법은? | list(), array(), numpy.array() |
| 다음 중 파이썬에서 배열의 크기를 얻는 방법은? | len() |
| 다음 중 파이썬에서 배열의 요소를 얻는 방법은? | [index] |
| 다음 중 파이썬에서 배열의 요소를 설정하는 방법은? | [index] = value |
| 다음 중 파이썬에서 배열의 요소를 추가하는 방법은? | .append(value) |
| 다음 중 파이썬에서 배열의 요소를 삭제하는 방법은? | .pop(index) |
| 다음 중 파이썬에서 배열의 요소를 정렬하는 방법은? | .sort() |
| 다음 중 파이썬에서 배열의 요소를 검색하는 방법은? | .index(value) |
| 다음 중 파이썬에서 배열의 요소를 슬라이싱하는 방법은? | [start:end] |

8. 파이썬에서 콜렉션 간략 정리

| 컬렉션 | 특징 | 예제 |
| --- | --- | --- |
| 리스트 | 순서가 있는 컬렉션으로, 인덱스를 통해 요소에 접근할 수 있습니다. | my\_list = [1, 2, 3] |
| 튜플 | 순서가 있는 컬렉션으로, 리스트와 달리 요소를 변경할 수 없습니다. | my\_tuple = (1, 2, 3) |
| 세트 | 순서가 없는 컬렉션으로, 중복된 요소가 없습니다. | my\_set = {1, 2, 3} |
| 딕셔너리 | 키와 값으로 이루어진 컬렉션으로, 키를 통해 값에 접근할 수 있습니다. | my\_dict = {'name': 'John Doe', 'age': 30} |

**2. 개발자가 만들어보는 프롬프트**개발자들이 자주 사용하는 코드를 프롬프트로 정의.   
구글링 하기 전에 머릿 속 생각을 프롬프트로 정의함

| No. | 프롬프트 - 문자열 관련 자주사용하는 것 |
| --- | --- |
| 1 | 동물들 = “강아지 고양이 닭 ”  빈공간을 기준으로 동물들의 문자를 구분하여 출력 |
| 2 | 동물들 = “강아지:고양이:닭 ”  “:”를 기준으로 동물들의 문자를 구분하여 출력 |
| 3 | 동물들 = “강아지:고양이:닭 ”  “:”를 기준으로 동물들의 문자를 구분하고 갯수를 출력 |
| 4 | 동물들 = “강아지 고양이 닭”  동물들에서 “닭”이 존재하는 위치값 |
| 5 | 동물들 = “강아지 고양이 닭 강아지 고양이 닭”  빈공간을 기준으로 값을 나눈다.  강아지 갯수, 고양이 갯수, 닭 개수를 출력한다 |
| 6 | 긴문자=”abcdefghijklmn” 긴문자에서 4번째 글자부터 나머지까지 잘라서 출력 |

| No. | 프롬프트 - 배열, 컬렉션 관련 자주사용하는 것 |
| --- | --- |
| 1 | 숫자들 = [1, 2, 3, 4, 5, 6, 7] 숫자들에서 3번째부터 6번째까지만 출력 |
| 2 | 동물들 = [강아지, 고양이, 강아지, 고양이, 강아지]  새로운동물들 = []  동물들의 값을   새로운둥물들에 값이 존재하지 않는다면  새로운동물들에 대입한다. |
| 3 | 다양한값들 = [1, 2, “고양이”, false, 13.5]  정수값들 = [] 다양한값들에서 정수값만 정수값들에 추가 |
| 4 | 점수 = [100, 99,89, 90, 10  합격 = [] 점수에서 80점 이상만 합격에 추가 |
| 5 | 점수 = [100, 99,89, 90, 10 ] 점수를 오름차순으로 정렬  점수를 내림차순으로 정렬 |
| 6 | 다양한값들 = [1, 2, false, 13.5] 다양한값들이 모두 숫자인지 채크 |
| 7 | 점수등급 = {“하급” : 70, 중급 : “80”, 고급: “90”} 점수등급에서 key와 value를 같이 출력한다 |
| 8 | 점수등급 = {“하급” : 70, 중급 : “80”, 고급: “90”} 점수등급에서 value < 90 것들만 key와 value를 출력한다 |
| 9 | 점수등급 = {“하급” : 70, 중급 : “80”, 고급: “90”} 점수등급에 key=“낙제”, value=60을 더한다.  점수등급에 key=”하급”을 삭제한다. |

| No. | 프롬프트 - 랜덤 자주사용하는 것 |
| --- | --- |
| 1 | 동물들 = [강아지, 고양이, 강아지, 고양이, 강아지]  동물들 중 랜덤하게 1개를 선택하여 출력 |
| 2 | 주사위숫자 =[1, 2, 3, 4, 5, 6] 주사위숫자 중 랜덤하게 1개 선택하여 출력 |
| 3 | 1부터 100까지 랜덤하게 1숫자른 후 출력하기를 5번 반복 |

**3. 파이썬에서 모듈과 패키지**

파이썬의 모듈과 패키지는 코드를 재사용하고 관리하기 위한 것입니다. 모듈은 파이썬 파일로, 코드와 데이터를 포함하고 있습니다. 패키지는 모듈의 모음으로, 파일 시스템의 디렉토리로 구성되어 있습니다.

파이썬에서 모듈과 패키지를 사용하는 방법은 다음과 같습니다.

* 모듈을 사용하려면 import문을 사용합니다.
* 패키지를 사용하려면 import문을 사용하여 패키지를 가져오고, from문을 사용하여 패키지의 모듈을 가져옵니다.

파이썬에서 모듈과 패키지를 사용하는 이점은 다음과 같습니다.

* 코드를 재사용할 수 있습니다.
* 코드를 관리할 수 있습니다.
* 코드를 테스트할 수 있습니다.
* 코드를 디버깅할 수 있습니다.
* 코드를 배포할 수 있습니다.

다음은 파이썬의 모듈과 패키지에 대한 10가지 개념입니다.

| 개념 | 설명 |
| --- | --- |
| 모듈 | 파이썬 파일로, 코드와 데이터를 포함하고 있습니다. |
| 패키지 | 모듈의 모음으로, 파일 시스템의 디렉토리로 구성되어 있습니다. |
| import문 | 모듈이나 패키지를 가져오는 문입니다. |
| from문 | 패키지의 모듈을 가져오는 문입니다. |
| 코드 재사용 | 모듈이나 패키지를 사용하여 코드를 재사용할 수 있습니다. |
| 코드 관리 | 모듈이나 패키지를 사용하여 코드를 관리할 수 있습니다. |
| 코드 테스트 | 모듈이나 패키지를 사용하여 코드를 테스트할 수 있습니다. |
| 코드 디버깅 | 모듈이나 패키지를 사용하여 코드를 디버깅할 수 있습니다. |
| 코드 배포 | 모듈이나 패키지를 사용하여 코드를 배포할 수 있습니다. |

*👉 개발자 note  
파이썬에서는 모듈과 패키지를 사용하는 .py 파일이 따로 존재해야 한다. 그러나 Colab에서는 파일이 하나다. 결과적으로 테스트를 위해 다음과 같이 하면된다. 강조하지만 주피터노트북과 Colab은 교육용이다. 현업에서는 수많은 소스파일들이 필요한 기능만 분리되어 사용되고 있다. 회사에서 사용하는 각종 문서양식과 마찬가지이다.*

Google Colaboratory(Colab)에서 모듈을 정의하고 import로 사용하는 방법은 다음과 같습니다.

1. Colab에서 새 셀을 생성합니다.
2. %%writefile filename.py 명령어를 사용하여 모듈을 정의합니다.
3. 모듈에 코드를 작성합니다.
4. %%python 명령어를 사용하여 모듈을 실행합니다.
5. 모듈에서 사용할 함수나 변수를 가져옵니다.
6. 모듈에서 사용할 함수나 변수를 사용합니다.

예를 들어, my\_module.py 모듈을 정의하려면 다음과 같은 코드를 사용할 수 있습니다.

Python

%%writefile my\_module.py

def square(x):

return x \* x

def cube(x):

return x \* x \* x

이 코드를 실행하면 my\_module.py 모듈이 정의됩니다. 이 모듈을 실행하려면 다음과 같은 코드를 사용할 수 있습니다.

Python

%%python

import my\_module

result = my\_module.square(5)

print(result)

이 코드는 다음과 같은 출력을 생성합니다.

코드 스니펫

25

Colab에서 모듈을 정의하고 import로 사용하는 방법에 대한 자세한 내용은 Colab 문서: https://colab.research.google.com/notebooks/intro.ipynb를 참조하세요.