

# 비주얼프로그래밍

## 과제 2.

20196014 고지흔

## 1번

### 문제해결방안

과제 1-7의 가위바위보를 RSP=("가위","바위","보")의 튜플로 두고 가위를 RSP[0] 바위를 RSP[1] 보를 RSP[2]로 변경했습니다.

### 소스코드

```
import random
rnd=user_win=com_win=user_lose=com_lose=0
RSP=("가위","바위","보")
print("%s%s 게임" % (RSP[0],RSP[1],RSP[2]))
print("컴퓨터 : ",com_win,"승",com_lose,"패,  당신 : ",user_win,"승",user_lose,"패",sep="")
while(user_win<3 and com_win<3):
    print("(라운드",rnd+1,")",sep="")
    rnd+=1
    computer = random.choice([RSP[0],RSP[1],RSP[2]])
    print("컴퓨터가 결정했습니다")
    user=input("무엇을 내시겠습니까? (%s, %s, %s) " % (RSP[0],RSP[1],RSP[2]))
    if((computer==RSP[0] and user==RSP[2]) or (computer==RSP[1] and user==RSP[0]) or (computer==RSP[2] and user==RSP[1])):
        print("컴퓨터는 ",computer,". 당신은 ",user,". 컴퓨터가 이겼습니다",sep="")
        com_win+=1
        user_lose+=1
        print("컴퓨터 : ",com_win,"승",com_lose,"패,  당신 : ",user_win,"승",user_lose,"패",sep="")
        continue

    if((computer==RSP[0] and user==RSP[1]) or (computer==RSP[1] and user==RSP[2]) or (computer==RSP[2] and user==RSP[0])):
        print("컴퓨터는 ",computer,". 당신은 ",user,". 당신이 이겼습니다",sep="")
        com_lose+=1
        user_win+=1
        print("컴퓨터 : ",com_win,"승",com_lose,"패,  당신 : ",user_win,"승",user_lose,"패",sep="")
        continue

    if((computer==RSP[0] and user==RSP[0]) or (computer==RSP[1] and user==RSP[1]) or (computer==RSP[2] and user==RSP[2])):
        print("컴퓨터는 ",computer,". 당신은 ",user,". 비겼습니다",sep="")
        print("컴퓨터 : ",com_win,"승",com_lose,"패,  당신 : ",user_win,"승",user_lose,"패",sep="")
        #무승부 기록은 ppt에 안 넣었길래 출력 안했습니다.
```

### 결과화면

```
가위바위보 게임
컴퓨터 : 0승 0패,  당신 : 0승 0패
(라운드1)
컴퓨터가 결정했습니다
무엇을 내시겠습니까? (가위, 바위, 보) 보
컴퓨터는 보, 당신은 보, 비겼습니다
컴퓨터 : 0승 0패,  당신 : 0승 0패
(라운드2)
컴퓨터가 결정했습니다
무엇을 내시겠습니까? (가위, 바위, 보) 바위
컴퓨터는 가위, 당신은 바위, 당신이 이겼습니다
컴퓨터 : 0승 1패,  당신 : 1승 0패
(라운드3)
컴퓨터가 결정했습니다
무엇을 내시겠습니까? (가위, 바위, 보) 가위
컴퓨터는 보, 당신은 가위, 당신이 이겼습니다
컴퓨터 : 0승 2패,  당신 : 2승 0패
(라운드4)
컴퓨터가 결정했습니다
무엇을 내시겠습니까? (가위, 바위, 보) 가위
컴퓨터는 가위, 당신은 가위, 비겼습니다
컴퓨터 : 0승 2패,  당신 : 2승 0패
(라운드5)
컴퓨터가 결정했습니다
무엇을 내시겠습니까? (가위, 바위, 보) 바위
컴퓨터는 보, 당신은 바위, 컴퓨터가 이겼습니다
컴퓨터 : 1승 2패,  당신 : 2승 1패
(라운드6)
컴퓨터가 결정했습니다
무엇을 내시겠습니까? (가위, 바위, 보) |
```

## 문제해결 방안

첫번째 형식에 월과 일을 01월 01일의 형식이 아니라 1월 1일의 형식으로 받아서 문자열을 쪼개는데 어려움이 있었습니다.

3가지 형식이 문자열 길이가 다르다는 것을 착안해 3가지 경우로 분리했습니다. 이 경우에서 이름이 3글자가 아닌 경우와 월 일이 2자리 숫자이거나 1자리 숫자인 경우를 고려해줬습니다.

모두 2000년생 이후에 태어났으므로 find 메소드를 이용해서 2를 찾고 이를 기준으로 슬라이싱 했습니다.

첫번째 형식은 년과 월 일의 위치도 슬라이싱 기준으로 고려해서 해결했습니다.

## 소스코드

```
intro=input("문장을 입력하세요 : ")
lenintro=len(intro)
if(lenintro>=31 and lenintro<=35): #*월 *일 ~ **월 **일까지 #이름 2글자 ~ 4글자까지 받음
    year=intro.find("년")
    month=intro.find("월")
    day=intro.find("일")
    print("분석된 생년월일: %s/%s/%s" % (intro[year-4:year], intro[year+2:month], intro[month+2:day]))
elif(lenintro>= 36 and lenintro<=38): #이름 2글자 ~ 4글자까지 받음
    print("분석된 생년월일: %s" % intro[intro.find("2"):intro.find("2")+10])
elif(lenintro>=25 and lenintro<=27): #이름 2글자 ~ 4글자까지 받음
    start=intro.find("2")
    print("분석된 생년월일: %s/%s/%s" %(intro[start:start+4],intro[start+4:start+6],intro[start+6:start+8] ))
else:
    print("형식에 맞지 않는 입력입니다. 프로그램이 종료됩니다.")
```

## 결과화면

```
--문장을 입력하세요 : 제 이름은 홍길동이고 2001년 10월 6일에 태어났습니다
분석된 생년월일: 2001/10/6
|
```

```
2001년 10월 6일
문장을 입력하세요 : 홍길동입니다. 생년월일은 20011006입니다
분석된 생년월일: 2001/10/06
|
```

```
--문장을 입력하세요 : 안녕하세요 홍길동입니다. 2001/10/06 이 제 생년월일입니다.
분석된 생년월일: 2001/10/06
|
```

3번

## 문제해결방안

+와 -기준으로 문자열을 잘라서 리스트로 만들었습니다 ex( ['245', '-12', '+154'] )

이를 for문을 통해 int형으로 바꾸면서 결과값인 sum에 더하는 형식으로 문제를 해결했습니다.

## 소스코드

```
equation=input("수식을 입력하세요 : ")
num=0
numed=0
numlist=[]
for i in equation:
    if(i=="+" or i=="-"):
        numlist.append(equation[numed:num])
        numed=num
        num+=1
numlist.append(equation[numed:]) #수식상 맨 마지막 항이 남아서 따로 넣어줌
sum=0
for i in numlist:
    sum=sum+int(i)
print("=" %d % sum)
```

## 결과화면

```
00014 > python 3.py
수식을 입력하세요 : 12+345+6789
= 7146
```

```
00014 > python 3.py
수식을 입력하세요 : 34-12+5
= 27
```

```
00014 > python 3.py
수식을 입력하세요 : 334+123-32+245-6-32
= 632
```

#### 4번

##### 문제해결방안

우선 이름을 key로 전화번호를 value로 딕셔너리를 만들었습니다.

key값만 리스트로 만들고 for문을 통해 리스트에 있는 문자열을 하나하나 가져온 후 if문을 통해 입력한 문자열이 리스트에 있는 문자열의 일부인 문자열인지 확인하는 방식으로 문제를 해결했습니다

입력값이 딕셔너리의 키 리스트에 없고 키 리스트의 일부가 아닌 경우 입력값이 add인지 딕셔너리의 리스트에 찾을 수 없는 입력값인지 확인해서 문제를 해결했습니다.

##### 소스코드

```
phone={"홍길동":'010-4444-5555',"김중앙":"'010-9191-8181','심청':"010-3232-5454"}
while True:
    namelist=list(phone.keys())
    name=input("이름>> ")
    passer=0
    for i in namelist:
        if name in i:
            print("%s %s" % (i,phone[i]))
            passer=1
            break
    if ((name not in namelist) and passer==0):
        if name=="add":
            newname=input("이름은? ")
            newphonenumber=input("전화번호는? ")
            phone[newname]=newphonenumber
            print("%s 전화번호가 추가되었습니다." % newname)
        else:
            print("찾을 수 없습니다")
```

##### 결과화면

```
00014_4.py ---
이름>> 홍길동
홍길동 010-4444-5555
이름>> 홍길
홍길동 010-4444-5555
이름>> 홍길서
찾을 수 없습니다
이름>> add
이름은? 강감찬
전화번호는? 010-5555-6666
강감찬 전화번호가 추가되었습니다.
이름>> 강감찬
강감찬 010-5555-6666
```

5번

## 문제해결방안

while 문을 통해 99999 이하의 숫자인지 확인했습니다

입력한 숫자의 길이가 4자리 이상인 경우에 문자열을 슬라이싱 하고 서로 더하는 방법을 통해 ,를 추가하였습니다.

딕셔너리에 123456789에 해당하는 한글과 대응시켜 키와 벨류를 만들고

또 다른 딕셔너리에 문자열을 거꾸로 더할거라서 -5 -4 -3 -2 -1를 각각 만 천 백 십의 자리에 대응시켜서 두 딕셔너리를 이용해 문자열을 불러오고 이 문자열 서로 더해가는 방식으로 문제를 해결했습니다

0과 십의 자리 이상의 1은 읽지 않으므로 생략해줬고 1의자리의 1만 따로 생각해주었습니다.

## 소스코드

```
num=input("숫자는? ")
while len(num)>5:
    print("99999이하의 수만 가능합니다")
    num=input("숫자는? ")

tmpnum=num
if len(tmpnum)>3:
    tmpnum=tmpnum[:len(tmpnum)-3]+" "+tmpnum[len(tmpnum)-3:len(tmpnum)]
print("%s 원" % tmpnum)

kornumdic={'1':'일', '2':'이', '3':'삼', '4':'사', '5':'오', '6':'육', '7':'칠', '8':'팔', '9':'구', '0':' '}
kornumdic2={-5:'만', -4:'천', -3:'백', -2:'십', -1:''}
kornumword=""
if num[-1]=="1":
    kornumword="일"

else:
    kornumword=kornumdic[num[-1]]

j=-1
while -j<len(num):
    if num[j-1]=="0":
        j-=1
        continue
    else:
        kornumword=kornumdic[num[j-1]]+kornumdic2[j-1]+kornumword
    j-=1
print("%s원" % kornumword)
|
```

## 결과화면

```
~~~~~
숫자는? 12345
12,345 원
만이천삼백사십오원
```

```
~~~~~
숫자는? 10006
10,006 원
만육원
```

6번

## 문제해결방안

주어진 apple를 리스트에 append 한 후 리스트에 다른 단어를 계속해서 append하는 방식으로 문제를 해결했습니다

단어가 3~6글자인지 확인 한 후

input받은 단어가 단어 리스트에 있는지 확인하고 있으면 1점 차감

없으면 리스트의 맨 마지막 문자열의 마지막 문자와 입력받은 문자열의 첫 문자가 같은 지 확인하고 같으면 1점을 추가 같지 않으면 1점을 차감하는 방식으로 문제를 해결했습니다

while문을 통해 10번째 경기 이후에 프로그램이 종료되게 했습니다.

## 소스코드

```
word=["apple"]
score=0
game=0
while game<10:
    print("단어(%s) : " % word[-1], end="")
    newword=input()
    while len(newword)>6 or len(newword)<3:
        print("3~6글자 단어만 입력해주세요")
        print("단어(%s) : " % word[-1], end="")
        newword=input()
    if newword in word:
        score-=1
        print("이미 나온 단어입니다. 1점 차감. 현재 점수 %d점" % score)
    elif newword[0]==word[-1][-1]:
        score+=1
        print("1점 획득. 현재 점수 %d점" % score)
        word.append(newword)
    else:
        score-=1
        print("끝말잇기가 안됩니다. 1점 차감. 현재 점수 %d점" % score)
    game+=1
```

## 결과화면

```
python 0.py
단어(apple) : ear
1점 획득. 현재 점수 1점
단어(ear) : radio
1점 획득. 현재 점수 2점
단어(radio) : audio
끝말잇기가 안됩니다. 1점 차감. 현재 점수 1점
단어(radio) : orange
1점 획득. 현재 점수 2점
단어(orange) : ear
이미 나온 단어입니다. 1점 차감. 현재 점수 1점
단어(orange) : |
```