

### 1. Ինչպե՞ս կարելի է ցուցակի տարրերի միջով անցնել և ցիկլ օգտագործելով Python-ում: (1 միավոր)

#### Պատասխան:

Python-ում ցուցակի տարրերի միջով անցնելու համար հաճախ օգտագործվում է for ցիկլը:

```
my_list = [1, 2, 3, 4, 5]
for item in my_list:
    print(item)
```

### 2. Ինչպե՞ս կարելի է ստուգել փոփոխականի տվյալների տիպը Python-ում: (1 միավոր)

#### Պատասխան:

type() ֆունկցիան օգտագործվում է փոփոխականի տվյալների տիպը ստուգելու համար:

```
x = 10
y = "Hello"
z = 3.14
```

```
print(type(x)) # Output: <class 'int'>
print(type(y)) # Output: <class 'str'>
print(type(z)) # Output: <class 'float'>
```

### 3. Բացատրեք ներքաշման դերը Python-ում: (1 միավոր)

#### Պատասխան:

Ներքաշումը Python-ում կարևոր է, քանի որ այն սահմանում է կոդի բլոկների կառուցվածքը: Այն որոշում է, թե որ հրամանները պատկանում են որոշակի բլոկին, ինչպիսիք են ֆունկցիայի մարմինները, ցիկլերը և պայմանական արտահայտությունները: Համակարգված ներքաշումը անհրաժեշտ է Python կոդի ճիշտ կատարման համար:

### 4. Բացատրեք, թե ինչպես են աշխատում if, elif և else արտահայտությունները օրինակով: (1 միավոր)

#### Պատասխան:

- **if:** Եթե պայմանը ճիշտ է, ապա կատարվում է if արտահայտության տակ գտնվող կոդի բլոկը:
- **elif:** Եթե նախորդ if և elif պայմանները սխալ են, ստուգվում է elif պայմանը: Եթե այն ճիշտ է, կատարվում է elif արտահայտության տակ գտնվող կոդի բլոկը:
- **else:** Եթե բոլոր նախորդ պայմանները սխալ են, կատարվում է else արտահայտության տակ գտնվող կոդի բլոկը:

#### Օրինակ:

```
x = 10
if x > 0:
    print("x is positive")
elif x == 0:
    print("x is zero")
else:
    print("x is negative")
```

### 5. Խնդրեք օգտատերին մուտքագրել ջերմաստիճանը Ցելսիուսով: Վերածեք այն ֆարենհայտի, օգտագործելով $F = C \times 9/5 + 32$ բանաձևը: (4 միավոր)

```
celsius = float(input("Enter temperature in Celsius: "))
fahrenheit = (celsius * 9/5) + 32
```

```
print("Temperature in Fahrenheit:", fahrenheit)
```

**6. Գրեք կարճ Python կոդ, օգտագործելով range կոնստրուկտորը, ստեղծելու ցուցակ արժեքներով [10, 20, 30, 40, 50, 60, 70]: (4 միավոր)**

```
my_list = list(range(10, 80, 10))  
print(my_list)
```

**7. Գրեք կարճ Python ֆունկցիա, minmax(data), որը վերցնում է մեկ կամ ավելի ոչ բացասական թվերի հաջորդականություն և վերադարձնում է ամենամեծ թիվը, որը բազմապատիկ է ամենափոքր թվի: Մի օգտագործեք ներկառուցված min կամ max ֆունկցիաները ձեր լուծումը իրականացնելիս: (Դիցուք, որ հաջորդականության բոլոր տարրերը մեծ են 1-ից:) (8 միավոր)**

```
def minmax(data):  
    smallest = data[0]  
    largest = data[0]  
    for num in data[1:]:  
        if num < smallest:  
            smallest = num  
        if num > largest:  
            largest = num  
    return largest // smallest * smallest
```