

## 1. Ինչպե՞ս կարելի է ստուգել տարրի գոյությունը հավաքածուում և բառարանում: (1 միավոր)

Պատասխան:

- **Հավաքածուներ (sets):**

- in օպերատորը օգտագործվում է տարրի գոյությունը ստուգելու համար:

```
my_set = {1, 2, 3}
if 2 in my_set:
    print("2 is in the set")
```

- **Բառարաններ (dictionaries):**

- in օպերատորը օգտագործվում է բանալու գոյությունը ստուգելու համար:

```
my_dict = {"apple": 1, "banana": 2}
if "apple" in my_dict:
    print("apple is in the dictionary")
```

## 2. Ի՞նչ է արդյունքը + օպերատորի օգտագործման դեպքում երկու ցուցակների վրա և + օպերատորի օգտագործման դեպքում ցուցակի և ամբողջ թվի վրա: (1 միավոր)

Պատասխան:

- **Երկու ցուցակների համար:** + օպերատորը կոնկատենացնում է երկու ցուցակները, ստեղծելով նոր ցուցակ, որը պարունակում է երկու ցուցակների բոլոր տարրերը:
- **Ցուցակի և ամբողջ թվի համար:** Այս գործողությունը Python-ում ուղղակիորեն չի աջակցվում: Դուք պետք է ամբողջ թիվը վերածեք մեկ տարր պարունակող ցուցակի և այնուհետև կոնկատենացնեք երկու ցուցակները:

## 3. 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")
```

## 4. Ի՞նչ է մեկնաբանությունների նպատակը Python կոդում: (1 միավոր)

Պատասխան:

Մեկնաբանությունները օգտագործվում են կոդը ավելի հասկանալի և պահպանելի դարձնելու համար: Դրանք չեն ազդում կոդի կատարման վրա: Մեկնաբանությունները կարող են օգտագործվել`

- Կողի նպատակը բացատրելու համար:
- Կողի տարբեր մասերի գործառույթը բացատրելու համար:
- Կողի հետագա պահպանման և փոփոխման համար:

**5. Գրեք ֆունկցիա max\_min, որը վերցնում է թվերի ցուցակ և վերադարձնում է ինչպես առավելագույն, այնպես էլ նվազագույն արժեքները: (4 միավոր)**

**Պատասխան:**

```
def max_min(numbers):
    if not numbers:
        return None, None
    max_num = min_num = numbers[0]
    for num in numbers[1:]:
        if num > max_num:
            max_num = num
        if num < min_num:
            min_num = num
    return max_num, min_num
```

**6. Գրեք ֆունկցիա, որը քառակուսացնում է տրված ցուցակի յուրաքանչյուր թիվը և տպում արդյունքում ստացված ցուցակը: (4 միավոր)**

**Պատասխան:**

```
def square_numbers(numbers):
    squared_numbers = [num**2 for num in numbers]
    print(squared_numbers)
```

**7. Գրեք Python ֆունկցիա, որը վերցնում է չկարգավորված զանգված և վերադարձնում է երկրորդ ամենամեծ տարրը: Ձեզ չի թույլատրվում կարգավորել զանգվածը: (8 միավոր)**

**Պատասխան:**

```
def second_largest(arr):
    if len(arr) < 2:
        return None
    largest = second_largest = float('-infinity')
    for num in arr:
        if num > largest:
            second_largest = largest
            largest = num
        elif num > second_largest and num != largest:
            second_largest = num
    return second_largest
```

- <https://github.com/sahil-kamleshwar/Python-Assignment>