

### **1. Բացատրեք հավաքածուների (sets) հասկացությունը Python-ում: (1 միավոր)** **Պատասխան:**

Հավաքածուները (sets) Python-ում ոչ կրկնվող տարրերի անկարգավոր հավաքածուներ են: Դրանք օգտագործվում են տարրերի գոյությունը արագ ստուգելու, կրկնությունները հեռացնելու և տարրերի միությունը, հատումը և տարբերությունը հաշվարկելու համար:

### **2. 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, պլեյերը և պայմանական արտահայտությունները: Համակարգված ներքաշումը անհրաժեշտ է Python կոդի ճիշտ կատարման համար:

### **3. Ինչպե՞ս կարելի է գրել բնադրված պայմանական արտահայտություն Python-ում: (1 միավոր)**

#### **Պատասխան:**

Բնադրված պայմանական արտահայտությունները Python-ում օգտագործվում են, երբ մեկ պայմանական արտահայտության ներսում գտնվում է մեկ այլ պայմանական արտահայտություն: Սա թույլ է տալիս ստուգել բազմակի պայմաններ և կատարել տարբեր գործողություններ, կախված այդ պայմանների ճշմարտության արժեքից:

#### **Օրինակ:**

```
x = 10
y = 5

if x > 0:
    if y > 0:
        print("Both x and y are positive")
    else:
        print("x is positive, but y is not")
else:
    print("x is not positive")
```

### **4. Ինչպե՞ս է աշխատում Python-ի ինտերպրետատորը: (1 միավոր)**

#### **Պատասխան:**

Python-ի ինտերպրետատորը կատարում է Python կոդը տող առ տող: Այն վերլուծում է յուրաքանչյուր տող, ստուգում է սինտաքսիսի սխալների համար և կատարում է հրամանները:

### **5. Օգտագործողից ստացեք տարրերի ցուցակը (բաժանված տարածությամբ) և տպեք նոր ցուցակ, որը պարունակում է միայն ներմուծված ցուցակի յուրօրինակ տարրերը, պահպանելով սկզբնական հերթականությունը: (4 միավոր)**

```
input_str = input("Enter elements separated by spaces: ")
input_list = input_str.split()
unique_list = []
for item in input_list:
    if item not in unique_list:
        unique_list.append(item)
print(unique_list)
```

### **6. Գրեք Python ֆունկցիա, որը վերցնում է տուփիկերի ցուցակ և վերադարձնում**

**Է նոր ցուցակ, որը կարգավորված է յուրաքանչյուր տուփի երկրորդ տարրի հիման վրա, օգտագործելով lambda ֆունկցիա: (4 միավոր)**

```
def sort_by_second_element(tuples):  
    return sorted(tuples, key=lambda x: x[1])
```

**7. Գրեք Python ծրագիր, որը տպում է բոլոր պարզ թվերը 1-ից 100 միջակայքում: (8 միավոր)**

```
def is_prime(num):  
    if num <= 1:  
        return False  
    for i in range(2, int(num**0.5) + 1):  
        if num % i == 0:  
            return False  
    return True  
  
for num in range(1, 101):  
    if is_prime(num):  
        print(num)
```

- <https://medium.com/@abelkrw/a-beginners-guide-to-getting-started-with-python-ffe0d788f1a2>
- [https://github.com/Neuro-kiran/Basic\\_Python\\_Codes](https://github.com/Neuro-kiran/Basic_Python_Codes)
- <https://github.com/monkmonk1152/python-doc-1>