

Here are the translations of the Python questions and their answers in Armenian:

Question 1: Ի՞նչ տարբերություն կա Python-ում while և for ցիկլերի միջև:

Answer:

- **While ցիկլ:**

- Կատարվում է այնքան ժամանակ, մինչև տրված պայմանը ճիշտ է:
- Պայմանը ստուգվում է յուրաքանչյուր իտերացիայից առաջ:
- Պետք է օգտագործել, երբ իտերացիաների քանակը նախապես անհայտ է:

Example:

```
i = 0
while i < 5:
    print(i)
    i += 1
```

- **For ցիկլ:**

- Իտերացիա է կատարվում արժեքների հաջորդականության վրա (օրինակ՝ ցուցակ, գույգ, տող կամ տիրույթ):
- Ցիկլը կատարվում է հաջորդականության յուրաքանչյուր տարրի համար:
- Պետք է օգտագործել, երբ իտերացիաների քանակը նախապես հայտնի է:

Example:

```
for i in range(5):
    print(i)
```

Question 2: Բացատրեք, թե ինչպես են աշխատում if, elif և else հրամանները օրինակով:

Answer:

- **if:** Կատարում է կոդի բլոկը, եթե պայմանը ճիշտ է:
- **elif:** Կատարում է կոդի բլոկը, եթե նախորդ if պայմանը սխալ է, և ընթացիկ elif պայմանը ճիշտ է:
- **else:** Կատարում է կոդի բլոկը, եթե բոլոր նախորդ if և elif պայմանները սխալ են:

Example:

```
x = 10

if x > 15:
    print("x is greater than 15")
elif x > 5:
    print("x is greater than 5 but less than or equal to 15")
else:
    print("x is less than or equal to 5")
```

Question 3: Ի՞նչ է կոմենտարների նպատակը Python կոդում:

Answer:

Կոմենտարները օգտագործվում են կոդի հատվածների նպատակը բացատրելու համար, դարձնելով այն ավելի հասկանալի և պահպանելի: Ինտերպրետատորը անտեսում է դրանք:

Example:

```
# This is a comment
x = 5 # This line assigns the value 5 to the variable x
```

Question 4: Ինչպե՞ս կարելի է ստեղծել անանուն \$ֆունկցիա Python-ում:

Answer:

Անանուն `ֆունկցիաները`, որոնք նաև հայտնի են որպես `lambda ֆունկցիաներ`, սահմանվում են `lambda` բառի միջոցով: Դրանք հաճախ օգտագործվում են կարճ, պարզ `ֆունկցիաների` համար:

Example:

```
square = lambda x: x * x
result = square(5)  # result will be 25
```

Question 5: Գրեք ծրագիր, որը ընդունում է ոչ բացասական ամբողջ թիվ `n` որպես մուտքային տվյալ և հաշվում `n`-ի ֆակտորիալը ցիկլի օգնությամբ: Տպեք արդյունքը:

```
n = int(input("Enter a non-negative integer: "))
factorial = 1

if n < 0:
    print("Factorial is not defined for negative numbers")
elif n == 0:
    print("The factorial of 0 is 1")
else:
    for i in range(1, n + 1):
        factorial *= i
    print("The factorial of", n, "is", factorial)
```

Question 6: Գրեք Python `ֆունկցիա`, որը ընդունում է բառերի ցուցակ և վերադարձնում նոր ցուցակ, որը պարունակում է միայն այն բառերը, որոնք սկսվում են բաղձույթով, յուրաքանչյուր բառը մեծատառով, օգտագործելով ցուցակային ըմբռնում:

```
def words_starting_with_vowel(words):
    vowels = ['A', 'E', 'I', 'O', 'U']
    return [word.capitalize() for word in words if word[0] in vowels]

words = ["apple", "banana", "orange", "elephant", "umbrella"]
result = words_starting_with_vowel(words)
print(result)  # Output: ['Apple', 'Elephant', 'Umbrella']
```

Question 7: Գրեք `ֆունկցիա`, որը ստեղծում է ֆիբոնաչի հաջորդականությունը մինչև `n` անդամ և վերադարձնում է այն որպես ցուցակ:

```
def fibonacci(n):
    if n <= 0:
        return []
    elif n == 1:
        return [0]
    else:
        fib_list = [0, 1]
        for i in range(2, n):
            fib_list.append(fib_list[i - 1] + fib_list[i - 2])
        return fib_list

n = 10
result = fibonacci(n)
```

```
print(result) # Output: [0, 1, 1, 2, 3, 5, 8, 13, 21, 34]
```

If you have any other questions or tasks, feel free to ask!

- <https://medium.com/@widqo756/control-flow-27b6435469aa>
- <https://github.com/visist16/Python-Programming>
- <https://github.com/Masumbiswas73/Assignment-DS>
- https://github.com/munga06/alx_python
- <https://wenku.csdn.net/column/4d5qxd8tfy>