1. **Print()** - ტერმინალზე გამოაქვს მითითებული არგუმენტი, იგივე ბეჭდვა. (ბრჭყალებში ჩაწერილი მონაცემები გამოაქვს უცვლელად, ხოლო ცვლადი ან ციფრების შედეგი თუ გვაინტერესებს ბრჭყალები არ ჭირდება)

მაგ: print(“hello, world!”) და print(5+10)

1. **=** - ტოლობით შეგვიძლია რაიმე ცვლადში შევინახოთ მონაცემი, ანუ მივანიჭოთ მნიშვნელობა.

მაგ: x ან y = 5 ან 10 - ს

შენიშვნა: ცვლადი არ შეიძლება დაიწყოს რიცხვით. მაგალითად 1x=5

რიცხვი შესაძლოა დავუწეროთ ცვლადის შემდეგ, მაგალითად: x1 ან x\_1

ასევე ორი ტოლობის დაწერისას შეგვიძლია შევადაროთ ორი მონაცემი ერთმანეთს True da False - ს პრინციპით.

1. **Int** – (ინგლისურად integer) - ნიშნავს მთელ რიცხვს. მაგ: x = 5
2. **Str** – (ინგლისურად string) - ნიშნავს ტექსტს. მაგ: name = “dato”
3. **\*\*** - ორი ფიფქი არის ახარისხების ბრძანება. ფიფქების შემდეგ ვწერთ ხარისხის ციფრს ან ცვლადს, რომლითაც ვახარისხებთ ძირითად რიცხვს ან ცვლადს.

მაგ: print(x\*\*y) ან print(5\*\*7)

1. **Float** - ათწილადი რიცხვი. მაგ: 3.0 ან 5.7
2. **Upper/lower** - ტექსტის დიდ ან პატარა ასოებზე გადაკეთება. იწერება ცვლადთან მიერთებული წერტილით.

მაგ: print(name.lower/epper())

1. **Type** - ფუნქციით მოწმდება რა ტიპის მონაცემია ცვლადში შენახული.

მაგ: print(type(name)) - თუ name-ში შენახულია სახელი, გამოიტანს string-ს, ციფრზე int - ს, ათწილადზე float.

1. **Bool** - ბულის ტიპის ნიშანი: True და False
2. **[ ]** - კვადრატულ ფორმის ფრჩხილებში შეგვიძლია შევიყვანოთ ლისტი(სია) სხვადასხვა ტიპის მონაცემებით. ლისტი არის ცვალებადი და შეგვიძლია მონაცემების ცვლილება.

მაგ: mylist = [1,2,3,”dato”]

1. **Len** – (ინგლისურად length-ზომა) - მონაცემების გაზომვა, დათვლა. რაოდენობრივად.

მაგალითად: len(mylist)

1. ინდექსირებით შესაძლებელია სიიდან კონკრეტული მონაცემის ამოღება თუ ვიცით, ინდექსირებით რომელ რიგითობაზე დგას მონაცემი. სიაში ინდექსირების(რიგითობის) ათვლა იწყება 0-დან. ინდექსის ამოღების დროს რიგითობის რიცხვი იწერება კვადრატულ ფრჩხილში

მაგ: print(mylist[3])

1. **.append** - სიაში რაიმეს ჩამატება. ეს კოდი იწერება ცვლადის შემდეგ გადაბმული წერტილით.

მაგ: mylist.append(“”)

1. **.pop/.remove** – ბრძანებები ავტომატურად ამოშლის სიაში ბოლო ადგილზე მყოფ ინდექს(მონაცემს), ხოლო ფრჩხილებში თუ ჩავუწერთ ინდექსის ნომერს წაშლის კონკრეტულად მითითებულ ინდექსს

მაგ: mylist.pop()

1. **{}** - ამ ტიპის ფრჩხილებში შეგვიძლია შევიყვანოტ სიმრავლე(სია). [] - გან განსხვავებიტ {} - მონაცემები არის არეული(დაულაგებელი). სიმრავლეში დაუშვებელია დუბლირებული მონაცემები. ასევე ლისტისგან განსხვავებით სიმრავლეშ დაუშვებელია მონაცემების შეცვლა/გადაწერა.

აღნიშნულ ფრჩხილებში ასევე იწერება ლექსიკონის ტიპის მონაცემები: key:value მაგ: cars = {“brand”:”mustang”, “model”:”mustang” } key და value გამოიყოფა ორწერტილით.

1. **.add** - სიმრავლეში რაიმეს ჩამატება. მაგ: myset(120)
2. **.union** - მონაცემების/სიმრავლეების გაერთიანება. მაგალითად თუ name - ს მივანიჭებ სახელებს და numbers - მივანიჭებ რიცხვებს. ამ სიის გაერთიანება შემიძლია ახალ სიმრავლეში, მაგ:

Numbers\_names = Numbers.union(name)

1. **.update** - ფუნქცია ახალი სიმრავლის შექმნის გარეშე ორ მონაცემებს აერთიანებს

Number.update(name)

1. **()** – (tuple) - მსგავს ფრჩხილებში ლისტის მსგავსად შეგვიძლია სხვადასხვა ტიპის მონაცემის შენახვა, ასევე დუბლიკატები. ტუპლში შენახული მონაცემები არ იცვლება, არ იშლება და არ ხდება ჩამატება. ტუპლში შენახვა უმჯობესია მაშინ, როცა მასში შეყვანილი მონაცემების შეცვლა აღარ იქნება საჭირო.
2. **.index** - ფუნქცია ეძებს ლისტში ან ტუპლში რიგითობით მერამდენეა ანუ ინდექის ნომრით მონაცემი. იწერება ცვლადის სახელთან მიბმული წერტილით.

მაგ: mylist.index(“???”)

1. **.get** - შეგვიძლია ლექსიკონიდან ამოვიღოთ კონკრეტული გასაღების მნიშვნელობა.

Key->value

მაგ: cars = {“brand”:”mustang”, “model”:”mustang” } ამ ლექსიკონიდან

Cars.get(“brand”)

1. **.keys** – ამ ფუნქციით შეგვიძლია ლექსიკონის გასაღებების დაბეჭდვა.

Cars.keys()

1. **.values** - იგივე პრინციპია მნიშვნელობაზე რაც გასაღებზე. ამით ვბეჭდავტ ლექსიკონის გასაღებების მნიშვნელობებს

Cars.values()

1. If/else/elif - ბრძანებით შესაძლებელია შევამოწმოთ რაიმე ლოგიკა, როგორც ექსელში.

If- უნდა გადავცეთ ლოგიკა და დავპრინტოთ სწორი პასუხის შემთვევაში, არასწორი პასუხის შემთხვევაში უნდა შემოვიტანოთ else ფუნქცია. Elif ფუნქცია შეგვიძლია გამოვიყენოთ სავარაუდო პასუხების გამოსატანად რამდენჯერაც გვინდა. ბოლოს ყოველთვის იწერება else და რჩება ცარიელი. დაპრინტვის დროს ფრჩხილებში ვუწერთ თუ რა გამოგვიტანოს ეკრანზე სიმართლის ან მცდარობის დროს. ეს კოდი იწერება ქვეშმიწერით და print-ს ჭირდება სივრცე კიდიდან.

მაგ: n=10

If n%2==0: *\*(თუ n გავყოთ 2ზე ტოლია ნოლის)*

Print(“ეს რიცხვი ლუწია”)

Elif: *\*(აქ შეგვიძლია დამატებით სავარაუდო პასუხები)*

Else:

Print(“ეს რიცხვი კენტია”)

1. **While/for loops** - ბევრი მონაცემის შესაყვანად/დასაბეჭდად. ახსნა რთულია დაგუგვლა ჯობს ☺