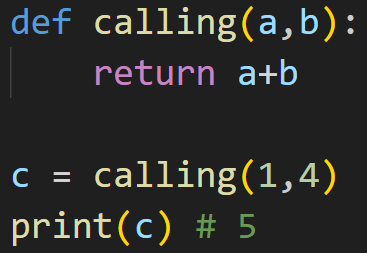
1. **הסברי בקצרה מהן פונקציות. מה הצורך בהן? מתי הן שימושיות?**

Functions are a piece of code that runs only when you call it.

The reason why we’d use functions, is if we want to make our code cleaner and if we have blots of code that are called muntiple times.

1. **הסברי בקצרה מהו ה – Value Return של פונקציה. תני דוגמא.**

In a function, the way to return a value is by the “return value”.



1. **.Pass by value / Pass by reference המושגים על בקצרה הסברי**

The concept “pass by reference” - is when you pass the variable name in the function and not a value itself, this means that if you did any changes in the function it would change the actually variable.

However , “pass by value” is when you pass a value in the funciton instead of a variable.

1. **הסברי בקצרה את המושג Scope ומה זמן חיים של משתנה/אובייקט (מתי הוא נוצר/ נהרס).   
   בתשובתך התייחסי להשפעת חלוקה לפונקציות ו-indentation.**

A scope is the term used to describe the kind of the variable. There are 4 kind of variables in the scope.

**Local** - a variable that was created inside a function and it’s lifespan is from the moment that funciton is in use untill the funciton finishes.

**Enclosing** - this is mainly if you have a nested function an you would like to use the same variable name in both funcitons, it’s lifespan is the same is a local vairable.

**Global** - a global variable is a variable that is alive for the duration of the whole code and can be accessed by everyone including in an import. That’s why you would normally only put necessary things in the global space (classes, functions)

**Built in** - built in words are words that are built in python and have special meanings (min, max..) they’re duration is during the whole code.

9. **רקורסיה:**

**a. הסברי בקצרה מהי רקורסיה.**

**b. מתי נרצה להשתמש ברקורסיה? האם ניתן לכתוב קוד ללא רקורסיה בכלל?**

**c. מה יעיל יותר מבחינת זמן ריצה, שימוש ברקורסיה או לולאות? הסבירי.**

1. Recursion is when you call a function from within the same function. The reason you would do that, is if you want to have the same function run over and over until it reaches a certain point. The no 1 must thing in recursion is that you have a value that will stop the recursion and that it changes from recursion to recursion.
2. I explained in ‘a’ when you would use recursion. Yes it’s possible to write code without recursion, all you need to do is not have the code call itself and just run once and in one direction.
3. If our main focus is run time than technically they have the same run time, but at the end of the dayt functions is better because in recursion you’re calling a function in each iteration meaning that it will take more time. However , it’s much easier to read recursions, it’s more compact and it looks more elegant.