python

Операторы

if	используется для проверки условий: если 1 условие верно 2 , выполняется блок выражений (называемый «if-блок»), иначе 3 выполняется другой блок выражений (называемый «else-блок»). Блок «else» является необязательным.
while	также является оператором цикла, который осуществляет итерацию по последовательности объектов, т.е. проходит через каждый элемент в последовательности.
for	позволяет многократно выполнять блок команд до тех пор, пока выполняется некоторое условие. Это один из так называемых операторов цикла. Он также может иметь необязательный пункт else .
break	служит для прерывания цикла, т.е. остановки выполнения команд даже если условие выполнения цикла ещё не приняло значения False или последовательность элементов не закончилась. Важно отметить, что если циклы for или while прервать оператором break, соответствующие им блоки else выполняться не будут.
continue	используется для указания Python, что необходимо пропустить все оставшиеся команды в текущем блоке цикла и продолжить 9 со следующей итерации цикла.
return	Оператор return используется для возврата из функции, т.е. для прекращения её работы и выхода из неё. При этом можно также вернуть некоторое значение из функции.

Функции

это многократно используемые фрагменты программы. Они позволяют дать имя определённому блоку команд с тем, чтобы впоследствии запускать этот блок по указанному имени в любом месте программы и сколь угодно много раз. Это называется вызовом функции.

local	При объявлении переменных внутри определения функции, они никоим образом не связаны с другими переменными с таким же именем за пределами функции – т.е. имена переменных являются локальными в функции. Это называется областью видимости переменной. Область видимости всех переменных ограничена блоком, в котором они объявлены, начиная с точки объявления имени.
global	Чтобы присвоить некоторое значение переменной, определённой на высшем уровне программы (т.е. не в какой-либо области видимости, как то функции или классы), необходимо указать Python, что её имя не локально, а глобально
nonlocal	тип области видимости, называемый «нелокальной» областью видимости, который представляет собой нечто среднее между первыми двумя. Нелокальные области видимости встречаются, когда вы определяете функции внутри функций.