```
"""
Topic: Python Keywords
@author: Varpe K. M.
Reference : Examples - https://www.geeksforgeeks.org/keywords-python-set-1/
"""
"""
try :
    This keyword is used for exception handling,
    used to catch the errors in the code using the keyword except.
    Code in "try" block is checked, if there is any type of error,
    except block is executed.

except :
    As explained above, this works together with "try" to catch exceptions.

raise :
    Also used for exception handling to explicitly raise exceptions.

finally :
    No matter what is result of the "try" block,
    block termed "finally" is always executed.

for :
    This keyword is used to control flow and for looping.

while :
    Has a similar working like "for" , used to control flow and for looping.

pass :
    It is the null statement in python.
    Nothing happens when this is encountered.
    This is used to prevent indentation errors and used as a placeholder

import :
    This statement is used to include a particular module into current program.

from :
    Generally used with import,
    from is used to import particular functionality from the module imported.

as :
    This keyword is used to create the alias for the module imported.
    i.e giving a new name to the imported module..
    E.g import math as mymath.


lambda :
    This keyword is used to make inline returning functions
    with no statements allowed internally

return :
    This keyword is used to return from the function.

yield :
    This keyword is used like return statement
    but is used to return a generator.

with : This keyword is used to wrap the execution of
    block of code within methods defined by context manager.
    This keyword is not used much in day to day programming.
```

```
in :
    This keyword is used to check if a container contains a value.
    This keyword is also used to loop through the container.

is : This keyword is used to test object identity, i.e
      to check if both the objects take same memory location or not.

global :
    This keyword is used to define a variable inside the function
    to be of a global scope.

non-local :
    This keyword works similar to the global, but rather than global,
    this keyword declares a variable to point to variable of
    outside enclosing function, in case of nested functions.
"""
```