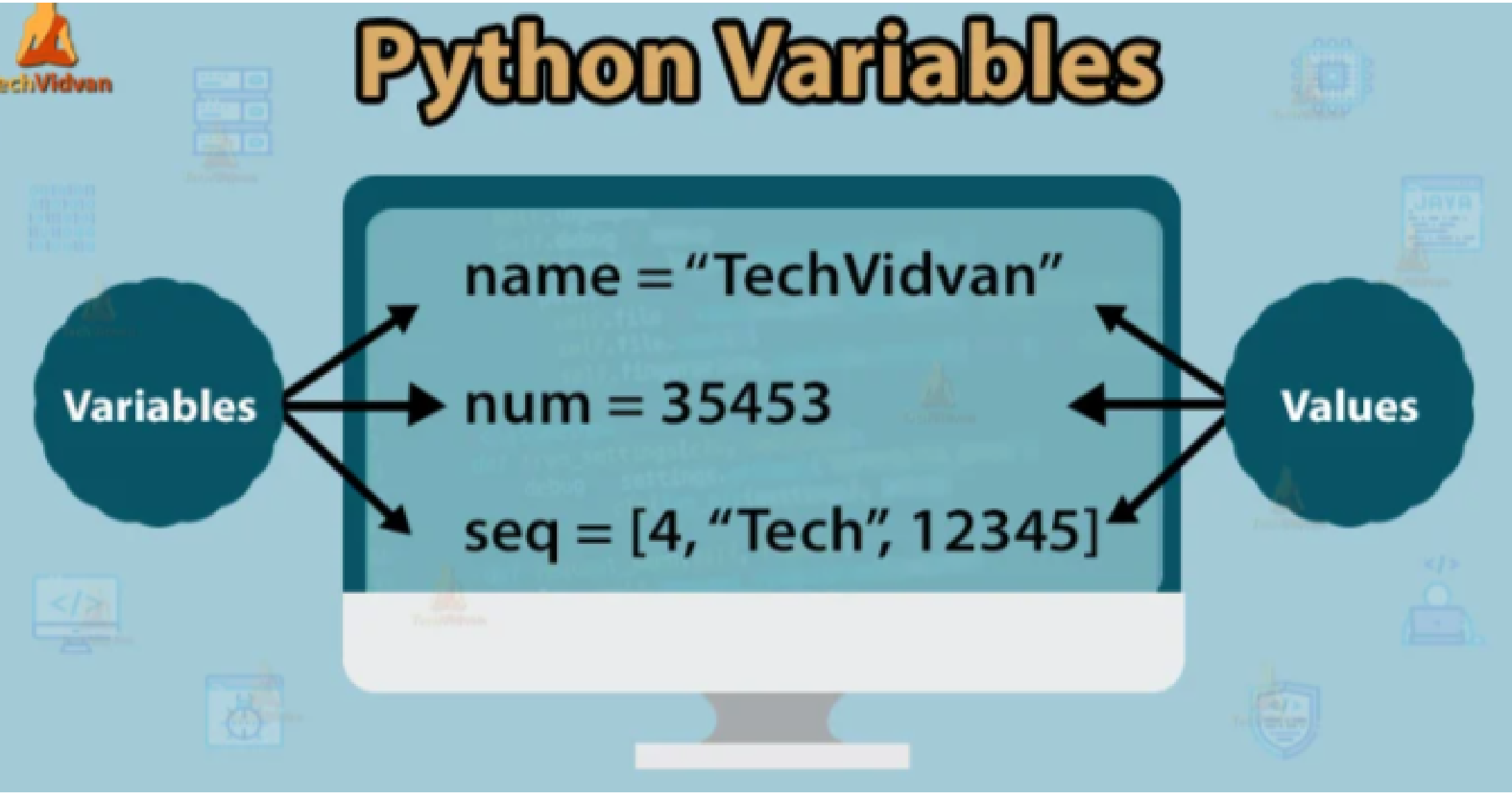
Python


Python Variables
Variables
name = "TechVidvan"
num = 35453
seq = [4, "Tech", 12345]
Values

VARIABLE TYPES

Variables are nothing but reserved memory locations to store values. This means that when you create a variable you reserve some space in memory.

#Assigning Values to Variables

The equal sign (=) is used to assign values to variables.

The operand to the left of the = operator is the name of the variable and the operand to the right of the = operator is the value stored in the variable.

EXAMPLE:

```
hp=764                  #An integer assignment
calorie=4.2             #A floating point
department='EEE'        #A string
print(hp,calorie,department )
```

RESULT

```
764 4.2 EEE
```

#Multiple Assignment

example

```
a,b,c=764,4.2,'EEE'
print(a,b,c)
```

Identifiers in python

Python identifiers are user-defined names.

They are used to specify the names of variables, functions, class, module, and objects.

Keywords in python

A python keyword is a reserved word which you can't use as a name of your variable, class, function, module and object.

These keywords have a special meaning and they are used for special purposes in Python programming language.

Python Keywords

Keywords are predefined, reserved words used in Python programming that have special meanings to the compiler.

We cannot use a keyword as a variable name, function name, or any other identifier. They are used to define the syntax and structure of the Python language.

All the keywords except True, False and None are in lowercase and they must be written as they are. The list of all the keywords is given below.

False	await	else	import	pass
None	break	except	in	raise
True	class	finally	is	return
and	continue	for	lambda	try
as	def	from	nonlocal	while
assert	del	global	not	with
async	elif	if	or	yield

Python Identifiers

Identifiers are the name given to variables, classes, methods, etc. For example,

```
language = 'Python'
```

Here, language is a variable (an identifier) which holds the value 'Python'.

We cannot use keywords as variable names as they are reserved names that are built-in to Python. For example,

```
continue = 'Python'
```

The above code is wrong because we have used continue as a variable name.

Rules for Naming an Identifier

1. Identifiers cannot be a keyword.
2. Identifiers are case-sensitive.
3. It can have a sequence of letters and digits. However, it must begin with a letter or _. The first letter of an identifier cannot be a digit.
4. It's a convention to start an identifier with a letter rather _.
5. Whitespaces are not allowed.
6. We cannot use special symbols like !, @, #, $, and so on.

Comments in Python

Comments in Python is the inclusion of short descriptions along with the code to increase its readability. A developer uses them to write his or her thought process while writing the code. It explains the basic logic behind why a particular line of code was written. They are just meant for the coders themselves or other developers to understand a piece of code, especially since the Python interpreter completely ignores comments in Python. You can see this in the following example.

There are multiple uses of writing comments in Python. Some significant uses include:

- Increasing readability
- Explaining the code to others
- Understanding the code easily after a long-term
- Including resources
- Re-using the existing code

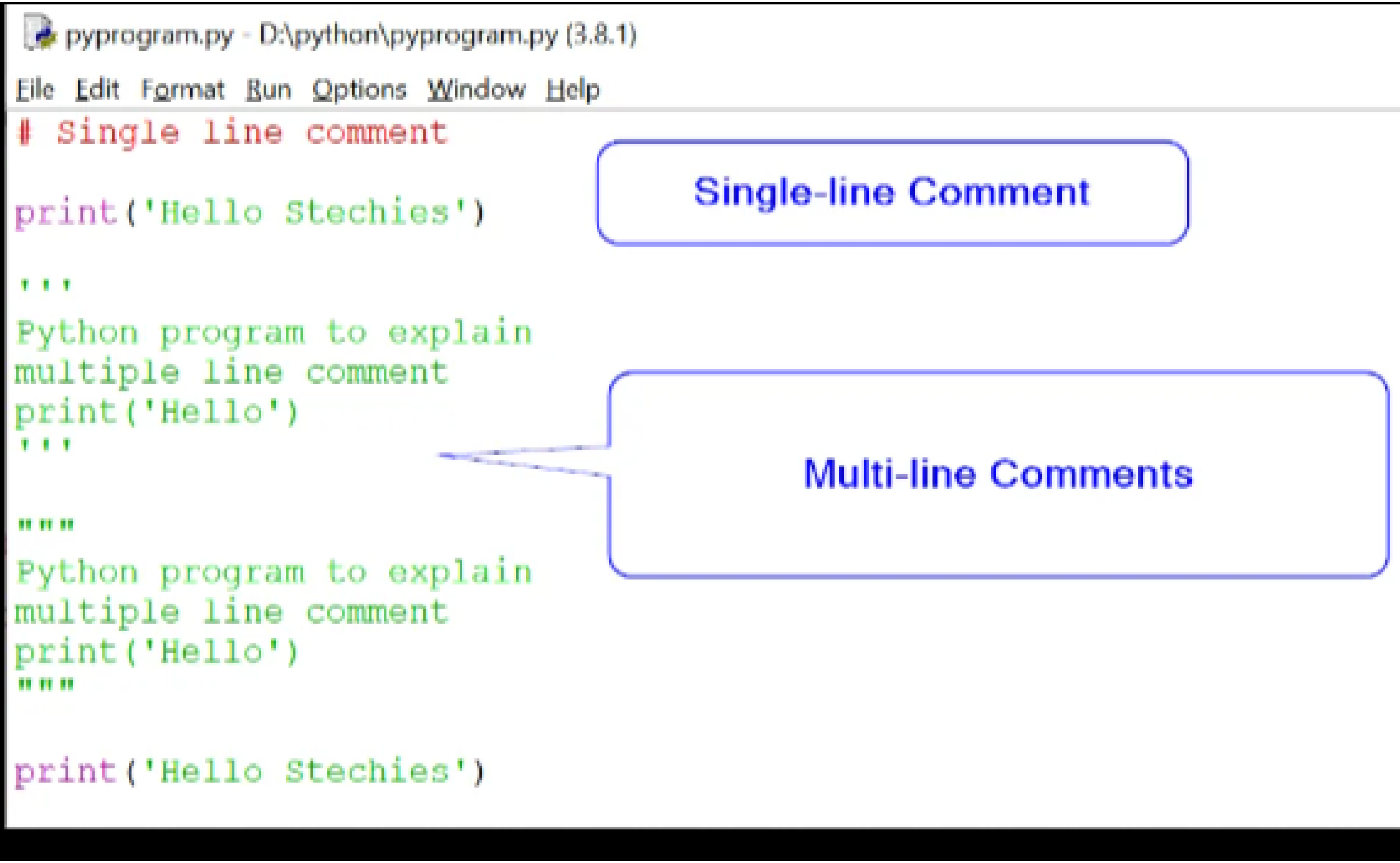
What Are the Advantages of Using Comments in Python?

Comments in Python provide numerous advantages. Their primary benefits include:

- Makes the code easily understandable by other programmers
- The code becomes self-explanatory
- Helps remember why we used a specific command, method, or function in the code
- Enables the interpreter to ignore some part of the code while testing

Python Comments

- To make the code much more readable.
- Python Interpreter ignores comment.
- Two types of comment is possible in python:
  - Single line comment and
  - Multi-line comment


pyprogram.py - D:\python\pyprogram.py (3.8.1)
File Edit Format Run Options Window Help
# Single line comment
print('Hello Stechies')
'''
Python program to explain
multiple line comment
print('Hello')
'''
"""
Python program to explain
multiple line comment
print('Hello')
"""
print('Hello Stechies')
Single-line Comment
Multi-line Comments

Escape sequence characters in Python

Escape sequences allow you to include special characters in strings. To do this, simply add a backslash (\) before the character you want to escape.

```
print ("Welcome to \'Clcoding\'")
```

```
Welcome to 'Clcoding'
```

\a	Alarm or Beep
\b	Backspace
\f	Form Feed
\n	New Line
\r	Carriage Return
\t	Tab (Horizontal)
\v	Vertical Tab
\\	Backslash
\'	Single Quote
\"	Double Quote
\?	Question Mark
\ooo	octal number
\xhh	hexadecimal number
\0	Null

```
print ("Welcome to \nClcoding")
```

```
Welcome to
Clcoding
```

```
print ("Welcome to \tClcoding")
```

```
Welcome to      Clcoding
```

```
print ("Welcome to ""Clcoding")
```

```
Welcome to Clcoding
```

```
print ("Welcome to \"Clcoding\"")
```

```
Welcome to "Clcoding"
```

```
print ("Welcome to 'Clcoding'")
```

```
Welcome to 'Clcoding'
```

```
print ("Welcome to \'Clcoding\'")
```

```
Welcome to 'Clcoding'
```