output → Python 3

```
print("_")
var=20
print(var)
print("value of var", var)
print("value of var" + "var2").
```

print("%s" % s)

%d - int

%s - str

Conversion	Meaning	Notes
'd'	Signed integer decimal.	
'i'	Signed integer decimal.	
'o'	Signed octal value.	(1)
'u'	Obsolete type – it is identical to 'd'.	(7)
'x'	Signed hexadecimal (lowercase).	(2)
'X'	Signed hexadecimal (uppercase).	(2)
'e'	Floating point exponential format (lowercase).	(3)
'E'	Floating point exponential format (uppercase).	(3)
'f'	Floating point decimal format.	(3)
'F'	Floating point decimal format.	(3)
'g'	Floating point format. Uses lowercase exponential format if exponent is less than -4 or not less than precision, decimal format otherwise.	(4)
'G'	Floating point format. Uses uppercase exponential format if exponent is less than -4 or not less than precision, decimal format otherwise.	(4)
'c'	Single character (accepts integer or single character string).	
'r'	String (converts any Python object using *repr()*).	(5)
's'	String (converts any Python object using `str()`).	(6)
'%'	No argument is converted, results in a '%' character in the result.	

3 -

print ("{0}, {1}".format(a, b, add))

4 print(f"{a}, {b}")

10 20