Sure! Here are the **30 MCQs on "Working with Functions in Python"** **(without answers)** – suitable for Class 12 Computer Science.

**🔹 MCQs: Working with Functions in Python**

**1.** What is the default return value of a function that doesn’t return anything explicitly?  
a) 0  
b) null  
c) None  
d) undefined

**2.** Which keyword is used to define a function in Python?  
a) function  
b) fun  
c) define  
d) def

**3.** What will be the output of the following code?

def add(x, y=2):

return x + y

print(add(3))

a) 5  
b) 3  
c) Error  
d) 2

**4.** Which of the following is NOT a valid parameter type in Python functions?  
a) Positional parameter  
b) Keyword parameter  
c) Static parameter  
d) Default parameter

**5.** What will be the output of the following code?

def f(a, L=[]):

L.append(a)

return L

print(f(1))

print(f(2))

a) [1], [2]  
b) [1], [1, 2]  
c) Error  
d) [2], [1, 2]

**6.** What does the \*args parameter allow in a function?  
a) Only one argument  
b) Arbitrary number of keyword arguments  
c) Arbitrary number of positional arguments  
d) Only integers

**7.** What is the result of calling locals() inside a function?  
a) All built-in functions  
b) List of modules  
c) Dictionary of local namespace  
d) List of global variables

**8.** What is the output of the following?

def func():

return

print(func())

a) None  
b) Error  
c) 0  
d) Empty string

**9.** What will the following function return?

def greet(name="Guest"):

return "Hello " + name

print(greet())

a) Hello  
b) Error  
c) Hello Guest  
d) Guest

**10.** Which of the following is used to return multiple values from a function?  
a) Array  
b) List  
c) Tuple  
d) Dictionary

**11.** What will be the output?

def outer():

x = 5

def inner():

return x + 1

return inner()

print(outer())

a) Error  
b) 6  
c) 5  
d) None

**12.** Which keyword is used to access a global variable inside a function?  
a) public  
b) global  
c) static  
d) extern

**13.** Which of the following function declarations is valid?  
a) def sum a, b:  
b) function sum(a, b):  
c) def sum(a, b):  
d) define sum(a b):

**14.** Which function returns the memory address of an object?  
a) id()  
b) type()  
c) mem()  
d) address()

**15.** In Python, parameters passed to functions are:  
a) Always copied  
b) Passed by value  
c) Passed by reference  
d) Passed by object reference

**16.** What is the correct order of function call resolution in Python?  
a) Built-in → Local → Global → Enclosing  
b) Local → Global → Enclosing → Built-in  
c) Local → Enclosing → Global → Built-in  
d) Global → Enclosing → Local → Built-in

**17.** Which function is used to get the name of the current module or function?  
a) \_\_name\_\_  
b) name()  
c) module()  
d) funcname()

**18.** Which statement is used to end the execution of a function and send a value back?  
a) break  
b) return  
c) end  
d) exit

**19.** What is the output?

def show():

print("Hello")

x = show()

print(x)

a) Hello, None  
b) None  
c) Hello  
d) Error

**20.** What does the following do?

def f(x): pass

a) Throws an error  
b) Defines an empty function  
c) Skips x  
d) Defines a class

**21.** What is the correct syntax to call a function named calc?  
a) call calc()  
b) calc[]  
c) calc()  
d) function calc()

**22.** Which of the following is used to define anonymous functions?  
a) def  
b) lambda  
c) nonlocal  
d) static

**23.** What does this code return?

def add(x, y):

return x + y

print(add("2", "3"))

a) 5  
b) 23  
c) "5"  
d) "23"

**24.** Which of the following function calls is invalid?  
a) f(2, 3)  
b) f(x=1, y=2)  
c) f(1, y=2)  
d) f(x=1, 2)

**25.** In Python, which function returns the number of arguments passed to a function at runtime?  
a) args()  
b) len(args)  
c) inspect.getargspec()  
d) count()

**26.** What will be the output?

def f():

print("Inside f")

x = f()

print(type(x))

a) <class 'NoneType'>  
b) <class 'function'>  
c) <class 'str'>  
d) Error

**27.** Can a function return another function in Python?  
a) Yes  
b) No

**28.** What does the following code print?

def calc(a, b):

return a + b

print(calc(b=4, a=5))

a) 9  
b) Error  
c) 45  
d) None

**29.** Which function is used to convert a string into a function object?  
a) str()  
b) lambda()  
c) eval()  
d) exec()

**30.** What is the output of the following code?

def square(x):

return x \* x

print(square(2 + 3))

a) 25  
b) 13  
c) 10  
d) 6

**31.** Which of the following function types is defined using the lambda keyword?  
a) Named function  
b) Recursive function  
c) Anonymous function  
d) Static function

**32.** In Python, default parameters:  
a) Must appear before non-default ones  
b) Can appear anywhere in the parameter list  
c) Must come after non-default parameters  
d) Are not allowed

**33.** What does the following function call print?

def show(msg, times=2):

print(msg \* times)

show("Hi")

a) Hi  
b) HiHi  
c) Error  
d) Hi2

**34.** Which symbol is used to define keyword arguments in function calls?  
a) \*  
b) :  
c) =  
d) @

**35.** What will the following function return?

def calc(x, y):

return x // y, x % y

a) A list  
b) A string  
c) A tuple  
d) Two separate integers

**36.** What is the scope of a variable declared inside a function?  
a) Global  
b) Class-level  
c) Local  
d) Static

**37.** What is the output of this recursive function?

def fact(n):

if n == 1:

return 1

return n \* fact(n-1)

print(fact(3))

a) 6  
b) 3  
c) 1  
d) 9

**38.** Which of the following statements is true?  
a) Python does not support variable number of arguments  
b) The \*args must come after \*\*kwargs  
c) The \*args must come before \*\*kwargs  
d) \*args and \*\*kwargs cannot be used together

**39.** What will be the output?

def greet(name="User"):

print("Welcome", name)

greet()

a) Welcome  
b) Welcome User  
c) Error  
d) Hello User

**40.** Which of the following will result in an error?

def f(a, b=1, c):

return a + b + c

a) Syntax is valid  
b) Error due to default parameter before non-default  
c) Works only with integers  
d) Returns tuple

**41.** What is the purpose of the return statement?  
a) Exit from a loop  
b) Call another function  
c) Exit a function and return a value  
d) Declare local variables

**42.** Which of the following functions is valid?

def f(x=1, y=2):

return x + y

a) Yes  
b) No  
c) Only for integers  
d) Only if y is a string

**43.** The function call print(func()) will print None if:  
a) func() contains a return statement  
b) func() has no return statement  
c) func() prints a value  
d) func() is not defined

**44.** What will be the output?

def test():

x = 100

test()

print(x)

a) 100  
b) Error: x is not defined  
c) None  
d) test

**45.** Which function would give you the names and values of all global variables?  
a) globals()  
b) locals()  
c) vars()  
d) globals(vars)

**46.** Which keyword can modify a variable from the outer (enclosing) function’s scope?  
a) global  
b) nonlocal  
c) static  
d) const

**47.** What does the pass keyword do in a function?  
a) Skips execution  
b) Terminates the program  
c) Exits the function  
d) Raises an exception

**48.** What is true about variable scope in functions?  
a) Local variables are automatically global  
b) Global variables can't be changed inside functions  
c) Local variables override global variables with the same name  
d) Global variables override locals

**49.** The output of the following function is:

def foo(x):

return x, x\*\*2

print(foo(3))

a) 9  
b) 3  
c) (3, 9)  
d) Error

**50.** What is required for a function to be recursive?  
a) Infinite loop  
b) While statement  
c) Base condition  
d) Default arguments

**51.** What happens when a function is called with more arguments than defined?  
a) Returns default values  
b) Ignores extra arguments  
c) Raises a TypeError  
d) Uses \*args automatically

**52.** When is a function said to be *pure*?  
a) It modifies global state  
b) It does not depend on or modify external state  
c) It uses I/O operations  
d) It calls another function

**53.** What is the output of:

def foo(x, y):

return x if x > y else y

print(foo(7, 12))

a) 7  
b) 12  
c) True  
d) False

**54.** What type of function does not use a name?  
a) Default function  
b) Inline function  
c) Anonymous function  
d) Static function

**55.** Which built-in function executes a string as code?  
a) str()  
b) exec()  
c) eval()  
d) code()

**56.** What does the following output?

def calc(x):

return x + 5

print(calc(3.0))

a) 8  
b) 8.0  
c) Error  
d) 35

**57.** What will be printed?

def func(a, b):

return a \* b

print(func("Hi", 3))

a) HiHiHi  
b) 333  
c) Hi 3  
d) Error

**58.** Which of the following defines a function with one required and one optional argument?  
a) def f(x=1, y):  
b) def f(x, y=1):  
c) def f(x=1, y=2=3):  
d) def f(x=1):

**59.** Which is a feature of Python functions?  
a) They can return only one value  
b) They can't have default values  
c) They can be nested  
d) They must always use return

**60.** What happens if a function has no return statement?  
a) Returns 0  
b) Returns undefined  
c) Returns None  
d) Throws an error

**61.** What is the correct way to call a function with both positional and keyword arguments?  
a) f(x=1, 2)  
b) f(1, y=2)  
c) f(x=1, y=2, 3)  
d) f(x, y)

**62.** Which function call will result in an error if the function is defined as def f(x, y=2):?  
a) f(3, 4)  
b) f(3)  
c) f()  
d) f(x=5)

**63.** Which of these defines a function with variable number of keyword arguments?  
a) def f(\*\*args):  
b) def f(\*kwargs):  
c) def f(args\*\*):  
d) def f(kwargs\*):

**64.** What is printed?

def test():

pass

print(type(test))

a) <class 'NoneType'>  
b) <class 'function'>  
c) function  
d) None

**65.** What is printed?

def add(x, y): return x + y

print(add(5, -2))

a) 7  
b) -3  
c) 3  
d) None

**66.** What will be the output?

def outer():

def inner():

return "Hello"

return inner

x = outer()

print(x())

a) Hello  
b) None  
c) inner  
d) Error

**67.** What does the following code return?

def foo(x, y):

return x + y

print(foo(y=2, x=3))

a) 5  
b) 32  
c) x + y  
d) Error

**68.** Which of these function headers is incorrect?  
a) def f(a, b):  
b) def f(a=1, b=2):  
c) def f(\*args, x):  
d) def f(x, y=1):

**69.** In a recursive function, what must always be present?  
a) return  
b) base condition  
c) parameter  
d) global variable

**70.** What is printed?

def f(a, b=2, c=3):

return a + b + c

print(f(1))

a) 6  
b) 5  
c) 3  
d) Error

**71.** What does this function return?

def multiply(a, b):

print(a \* b)

a) Value of a\*b  
b) None  
c) Error  
d) True

**72.** What happens if a variable is not found in local, enclosing, or global scope?  
a) Returns default  
b) Looks in built-ins  
c) Returns zero  
d) Raises NameError

**73.** What is printed?

def a(): return 1

def b(): return 2

print(a() + b())

a) 1  
b) 2  
c) 3  
d) Error

**74.** How many arguments can \*args accept?  
a) Only 1  
b) Up to 5  
c) Infinite positional arguments  
d) Only strings

**75.** What happens if you forget parentheses when calling a function?  
a) Syntax error  
b) Nothing, it prints function object  
c) Calls function  
d) Returns zero

**76.** What is the output?

def greet(name):

return "Hi " + name

print(greet("Ravi"))

a) HiRavi  
b) Ravi  
c) Hi Ravi  
d) Error

**77.** What will return without a value return?  
a) 0  
b) null  
c) None  
d) Empty string

**78.** What is required after the def statement?  
a) A colon :  
b) A semicolon ;  
c) Curly braces  
d) Brackets

**79.** What does \*args return inside a function?  
a) List  
b) Dictionary  
c) Tuple  
d) Set

**80.** What is printed?

def say(): return "Hi"

print(say)

a) Hi  
b) Error  
c) Reference to function  
d) None

**81.** Which is not a valid function name?  
a) func\_1  
b) 2func  
c) myFunc  
d) \_temp

**82.** What does this function do?

def dummy(): pass

a) Creates placeholder function  
b) Returns error  
c) Skips execution  
d) Deletes memory