Unit 1 · Bigger Programs

WORKSHEET

Full marks 32

Name _____ Class _____ Roll no. ____ Date _____ Marks _____ / 32

1 Key words [4]

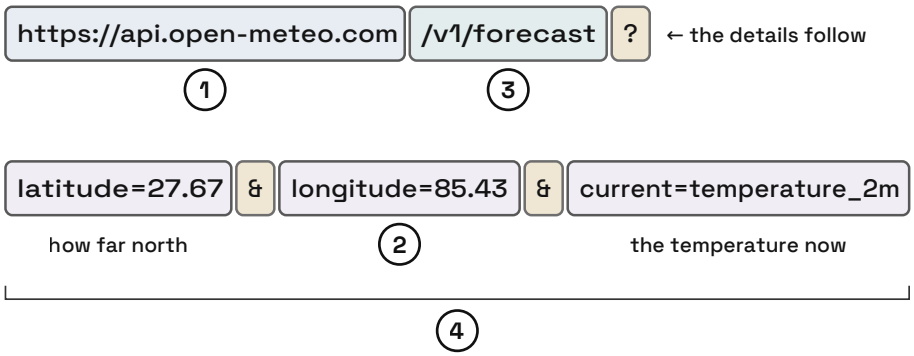
Fill in the blanks with words from the box. Two words in the box are not needed. Each blank carries 0.5 mark.

API	**docstring**	**JSON**	**library**	**module**	**namespace**	**pip**	**pure function**	**scope**
side effect								

A function that only takes values in and gives an answer back with `return` is a _____. Anything else a function does, such as printing or writing a file, is a _____. The part of a program where a name can be used is the name's _____. A `.py` file whose functions other programs can import is a _____, and a collection of modules written for other programs to use is a _____. Each module has its own box of names, its _____. The published way for one program to ask another for something is its _____, and most web APIs answer in _____.

2 Label the diagram [2]

The diagram shows a request to a weather API. Write what each part is for in place of 1 to 4. Each carries 0.5 mark.



1. _____ 2. _____

3. _____ 4. _____

3 Multiple choice [5]

Choose the correct option for each of the following questions.

i. Which of these is a **pure function**? [1]

a. a function that prints a bill
b. a function that takes a price and a number of plates of momo and returns the total
c. a function that adds a new student to a list kept outside it
d. a function that saves the marks to a file

ii. In top-down design, what do you write first? [1]

a. the smallest functions, so the main program can use them

b. the main steps of the program, as calls to functions not yet written

c. the tests

d. the documentation of the libraries you will use

iii. A program has `rate = 13` at the top, outside every function. A function contains only `return price * rate / 100`. What happens when it is called? [1]

a. It uses the global rate, 13.

b. NameError: the function can't see rate.

c. UnboundLocalError.

d. It uses 0, because rate isn't one of its parameters.

iv. A function's documentation says `round_to(x, *, places=0)`. Which call is correct? [1]

a. round_to(3.14159, 2)

b. round_to(3.14159, places=2)

c. round_to(*, places=2)

d. round_to(places=2)

v. An API answers `{"district": "Ilam", "tea_gardens": [{"name": "Kanyam", "area": 85}]}`, loaded into **data**. Which reaches the 85? [1]

a. `data["area"]`

b. `data["tea_gardens"]["area"]`

c. `data["tea_gardens"][0]["area"]`

d. `data[1][0][1]`

4 What does it print? [6]

Write exactly what each program prints, in the box under it.

a. [3]

```
1  total = 100
2
3  def spend(amount):
4      total = 50
5      return total - amount
6
7  def show():
8      return total + 1
9
10 print(spend(20))
11 print(show())
12 print(total)
```

OUTPUT

b. [3]

```
def top_two(marks):
    marks.sort(reverse=True)
    return marks[:2]

science = [45, 72, 21, 63]
best = top_two(science)
print(best)
print(science)
print(sorted(science)[0], len(best))
```

OUTPUT

5 Trace the program [4]

Fill in one row for each time round the loop in `tally`, then write what the program prints. The first row shows the values before the loop.

```
GRADES = [(90, "A+"), (80, "A"), (70, "B+"), (60, "B"),
          (50, "C+"), (40, "C"), (35, "D")]

def grade_for(p):
    for lowest, grade in GRADES:
        if p >= lowest:
            return grade
    return "NG"

def tally(percents):
    counts = {}
    for p in percents:
        g = grade_for(p)
        if g in counts:
            counts[g] = counts[g] + 1
        else:
            counts[g] = 1
    return counts

print(tally([72, 53, 92, 75, 33]))
```

p	g	counts[g] after the round
(before)		

OUTPUT

6 Find and fix the bug [3]

Read the program, then answer.

A museum's program should count its visitors, but it stops with an error on line 6.

```
1  count = 0
2
3  def add_visitors(people):
4      count = count + people
5
6  add_visitors(12)
7  add_visitors(30)
8  print(count)
```

i. Which line really causes the error, and what is the error called?
ii. Why does it happen?
iii. Rewrite the program so that `add_visitors` is a pure function. It should print 42.

7 Write a program [4]

Write the program in Python on the lines. Any program that gives the right result gets the marks.

A variable **text** holds this answer from an API, as a string:

```
{"province": "Karnali",
 "lakes": [{"name": "Rara", "area_km2": 10.8},
           {"name": "Phoksundo", "area_km2": 4.9}]}
```

Write a program that turns it into Python data, prints each lake's name and area, one lake a line, and then prints the name of the largest lake. It should work however many lakes the answer has.

8 Short answer [2]

Write a short answer to the following question.

A file **shop.py** ends with these lines. What do they do when shop.py is run by itself? And when a program runs `import shop`?

```
if __name__ == "__main__":
    print(bill_total(150, 2))
```

9 Think [2]

Think, then answer.

Ms. Karki's **marks_file.py** was rewritten to read the file with the **csv** module instead of `split(",")`, yet her main program and **report.py** didn't change a single line. Why didn't they need to? Why does this matter even more in a program of thousands of lines?

10 Activity: Ask the districts API (not marked)

At a computer, open the unit's page (the QR code) and find **Ask the districts API.** Send each request, and write the status and what you found.

Request (after .../nepal/v1)	Status	What came back
`/provinces`		How many districts has Karnali?
`/districts?province=Karnali`		The first district in the list:
`/districts/Bhaktapur`		Headquarters and population:
`/districts/Baktapur`		
`/districts?province=Karnaly`		
`/districts/` and your own district		