Quiz 08

Due Nov 8 at 10pm **Points** 10 **Questions** 5 **Time Limit** None

Instructions

Answer the following questions in your own words. Do NOT simply cut and paste the information from the slides, the web, or any other source. You will receive a score of 0 if you copy the prose from any source.

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	23 minutes	10 out of 10

Score for this quiz: **10** out of 10

Submitted Nov 8 at 8:26pm

This attempt took 23 minutes.

Question 1

2 / 2 pts

Given

```
def foo(x, use_max = True, *values):
    return x + (min(values) if use_max else max(values))
```

```
>>> foo(1, 2, 3, 4) # call foo()
```

1. What is the value of the use_max?

2. What is the value of 'values'?

3. What value is returned from the function call? Why?

Your Answer:

1. The value of use_max is 2 since it is the second parameter

2. The value of 'values' = (3,4)

3. Since x takes the value of1 & use_max evaluates to true so we will be adding 1 to the minimum value of 'values'(3,4) which is 1+3 = 4

1. use_max is assigned the value of the second parameter so use_max == 2

2. values == (3, 4)

3. use_max evaluates to True so the expression becomes 1 + min(3, 4) == 4

Question 2

2 / 2 pts

Python provides three different ways to import modules:

import module_name as alias

from module_name import objects

from module_name import *

Why should you avoid "from module_name import *"?

Your Answer:

"from module_name import*" can raise many conflicts since it imports all the functions present in the module so if there is an existing function name that matches any of your variable or function names then the conflict will occur. We will lose control over what to include and what not to.

import * may redefine important function definitions with potentially dangerous alternatives that can cause unexpected results.

Question 3

2 / 2 pts

Write a function that demonstrates a good use of using keyword parameters. Don't use the examples from the lecture.

Your Answer:

```
def professors(**kwargs):

    for key, value in kwargs.items():

        print(f"{key} is a professor at {value}")

professors(Raz = Stevens)
```

Output:

```
Raz is a professor at Stevens
```

Example from the lecture:

```
def keywords(**kwargs):
    for key, value in kwargs.items():
        print(f"{key}={value}")

keywords(name="jim", age="ancient", happy=True)
```

Question 4

2 / 2 pts

You are starting a project with several developers to write a large new Python program. Explain how you can use modules to organize the code for your project.

Say your project has file1.py, file2.py, file3.py, and file4.py. File2.py needs class foo from file3.py and class bar from file4.py. Write the Python code that you would include in file2.py to make foo and bar available.

Your Answer:

Modules can be used to organize the code based on different functionalities of the project. Eg: login functionality can have a different module, billing functionality can have a different module. We can reuse the code of one module into another module when required.

In order to make foo and bar available in file2.py we need to import foo from file3.py and bar from file4.py at the beginning of file2.py

from file3 import foo

from file4 import bar

file2.py should include the following lines:

from file3 import foo

from file4 import bar

Question 5

2 / 2 pts

How can pypi.org help to make Python programmers more productive?

Your Answer:

pypi.org helps developers to find and install relevant packages and solutions already available. One can publish the packages they designed too. One can distribute their software using pypi.org. pypi helps in building a community for python developers where they can learn share and grow together.

Rather than solving all problems from scratch, Pythonistas can look to PyPI.org to see if existing solution already exist.

Quiz Score: **10** out of 10