



Week 4 Exercise

测验, 10 个问题

1
point

1.
Consider this code:

```
1 class Contact:
2     """ A contact with a first name, a last name, and
3         an email address. """
4     def __init__(self, first_name, last_name,
5                   email_address):
6         """ (Contact, str, str, str) -> NoneType
7
8         Initialize this Contact with first name
9         first_name, last name
10        last_name, and email address email_address.
11        """
12        self.first_name = first_name
13        self.last_name = last_name
14        self.email_address = email_address
```

Select the code fragment(s) that create and initialize a **Contact** using the constructor (method `__init__`).



```
1 paul = Contact('Paul', 'Gries',
2               'paul@example.com')
```



```
1 contact = Contact()
2
3 paul = Contact(contact, 'Paul', 'Gries',
4               'paul@example.com')
```



```
1 paul = Contact()
2
3 paul.first_name = 'Paul'
4
5 paul.last_name = 'Gries'
6
7 paul.email_address = 'paul@example.com'
```



```
1 info = ['Paul', 'Gries', 'paul@example.com']
2
3 paul = Contact(info)
```

1
point

2.
This question uses class **Contact** from the previous question.

Variable **jen** refers to a **Contact** object. Select the correct way to print **jen**'s email address.

- ☒ `print(jen.email_address)`
- ☐ `print(jen[2])`
- ☐ `print(self.email_address)`
- ☐ `print(jen.self.email_address)`

1
point



Week 4 Exercise

This question uses class **Contact** from the previous questions.

测验, 10 个问题

Another method has been added to class **Contact** :

```
1 def add_phone_number(self, telephone_num):
2     """ (Contact, str) -> NoneType
3
4     Add phone number telephone_num for this contact.
5     """
6
7     self.phone_number = telephone_num
```

For a variable **khaled** that refers to a **Contact** object, which code fragment correctly calls method **add_phone_number** ?

- ☐ `khaled.add_phone_number(khaled, '555-1111')`
- ☐ `khaled.add_phone_number() = '555-1111'`
- ☒ `khaled.add_phone_number('555-1111')`
- ☐ `add_phone_number(khaled, '555-1111')`

1
point

4.

This question uses class **Contact** from the previous questions, and also uses types **str**, **float**, and **list**.

Here are several code fragments. In each fragment, there is a pair of method calls. In some pairs, the two method calls are equivalent to each other, and in the others, the two method calls are not equivalent to each other. Select the code fragment(s) in which the method calls are equivalent to each other.

Assume that variable **c** refers to a **Contact** and that variable **L** refers to a **list**.

☐

```
1 (0.6).as_integer_ratio()
2
3 float.as_integer_ratio(float, 0.6)
```

☒

```
1 str.replace('abc 123', '123', '246')
2 'abc 123'.replace('123', '246')
```

☐

```
1 c.add_phone_number('555-1111')
2
3 c.add_phone_number(c1, '555-1111')
```

☐

```
1 L.index(3)
2
3 list.index(3)
```

☒

```
1 c.add_phone_number('555-1111')
2
3 Contact.add_phone_number(c, '555-1111')
```

1
point

5.

This question uses class **Contact** from the previous questions.

Variable **rorik** refers to a **Contact** object with instance variables **first_name**, **last_name** and **email_address** that refer to **'Rorik'**, **'Henrikson'** and **'rorik@example.com'** respectively.

What is produced when **str(rorik)** is called?

- ☒ A string containing information about the object that **rorik** refers to. This string contains both its type and its memory address.



Week 4 Exercise

A string containing the types and memory addresses of the objects that **first_name**, **last_name**, and **email_address** refer to.

测验, 10 个问题

- ☐ 'Rorik Henrikson <rorik@example.com>'
- ☐ 'Henrikson, Rorik <rorik@example.com>'

1
point

6.

This question uses class **Contact** from the previous questions.

Another method has been added to class **Contact** :

```
1 def __str__(self):
2     """ (Contact) -> str
3
4     Return a string representation of this contact.
5     """
6
7     return '{0} {1} <{2}>'.format(self.first_name,
8                                   self.last_name, self.email_address)
```

Variable **rorik** refers to a **Contact** object with instance variables **first_name**, **last_name**, and **email_address** that refer to 'Rorik', 'Henrikson' and 'rorik@example.com' respectively.

What is produced when **str(rorik)** is called?

- ☐ A string containing the types and memory addresses of the objects that **last_name**, **first_name**, and **email_address** refer to.
- ☒ 'Rorik Henrikson <rorik@example.com>'
- ☐ 'Henrikson, Rorik <rorik@example.com>'
- ☐ A string containing information about the object that **rorik** refers to. This string contains both its type and its memory address.

1
point

7.

This question uses class **Contact** from the previous questions.

Consider this code:

```
1 class Email:
2     """ An email with a list of recipients, a subject
3         and a body. """
4
5     def __init__(self, recipients, subject, body):
6         """ (Email, list of Contact, str, str) ->
7             NoneType
8
9             Initialize this Email with recipients, subject
10            and body. """
11
12            self.recipients = recipients
13            self.subject = subject
14            self.body = body
```

Which of the following can be used to create an **Email** object?

☐

```
1 new_email = Email()
```

☐

```
1
```



Week 4 Exercise 8

测验, 10 个问题

```
1 new_email = Email('Hello', 'Hi there!\n Bye  
2 student1 = Contact('Hugh', 'Z.',  
3 hugh@fakedomain.com')  
4  
5 student2 = Contact('Kathryn', 'Z.',  
6 kathryn@fakedomain.com')  
7  
8 student3 = Contact('Karin', 'Z.',  
9 karin@fakedomain.com')  
10  
11 students = [student1, student2, student3]  
12  
13 subject = 'LTP2: E4 is posted!'  
14  
15 body = 'Hello,\nE4 is posted. Good luck!\n  
16 Paul and Jen'  
17  
18 new_email = Email(students, subject, body)
```



```
1 new_email = Email([Contact('Kathryn', 'Z.',  
2 kathryn@fakedomain.com')], 'Hello', 'Hi  
3 there!\n Bye for now.')
```

1
point

8.

This question uses classes **Contact** and **Email** from the previous questions.

This method is added to class **Email** :

```
1 def __str__(self):  
2     """ (Email) -> str  
3  
4     Return a string representation of this email.  
5     """  
6  
7     result = 'To: '  
8     for contact in self.recipients:  
9         result = result + '{0}, '.format(contact)  
10  
11     result = result + '\nSubject: {0}'.format(self.subject)  
12     result = result + '\n{0}'.format(self.body)  
13     return result
```

Variable **message** refers to an **Email** object created with:

- recipients:
`[Contact('Paul', 'Gries', 'paul@example.com'), Contact('Jen', 'Campbell', 'jen@example.com')]`
- subject: `'2nd MOOC'` , and
- body `'Hi!\nI hope your 2nd MOOC is going well!\nBye :-)'` .

What is printed when `print(message)` is executed?



To: [Paul Gries <paul@example.com>, Jen Campbell <jen@example.com>]

Subject: 2nd MOOC

Hi!

I hope your 2nd MOOC is going well!

Bye :-)



To: Contact('Paul', 'Gries', 'paul@example.com'), Contact('Jen', 'Campbell', 'jen@example.com')

Subject: 2nd MOOC

Hi!

I hope your 2nd MOOC is going well!

Bye :-)



To: [Contact('Paul', 'Gries', 'paul@example.com'), Contact('Jen', 'Campbell', 'jen@example.com')]



Week 4 Exercise

Subject: 2nd MOOC
测验, 10 个问题

I hope your 2nd MOOC is going well!

Bye :-)

☒ To: Paul Gries <paul@example.com>, Jen Campbell <jen@example.com>,

Subject: 2nd MOOC

Hi!

I hope your 2nd MOOC is going well!

Bye :-)

1
point

9.
Which of the following is **not** a special method of **object** ?

- ☐ `--ne--`
- ☒ `--lower--`
- ☐ `--str--`
- ☐ `--eq--`

1
point

10.
Consider this code:

```
1 class Author:
2     def __init__(self, name):
3         """ (_____, str) -> NoneType """
4         self.name = name
```

What should the blank (_____) in the type contract be replaced with?

- ☒ `Author`
- ☐ `str`
- ☐ It is not possible to tell.
- ☐ `NoneType`

☒ 我（**伟臣 沈**）了解提交不是我自己完成的作业 将永远不会通过此课程或导致我的 Coursera 帐号被关闭。
了解荣誉准则的更多信息

提交测试



Week 4 Exercise

测验, 10 个问题