Week - 7, Graded, Theory

Solution
Question-17
Answer
Solution

Problem-1

Questions 1 to 4 are based on common theme

Use the data block given below to answer questions 1 to 4.

```
org_role_list = [("dev", "developer"),
                 ("ba", "business analyst"),
                 ("qa", "quality analyst"),
                 ("lead", "program manager"),
                 ("ra", "research associate")]
org_work = {0: "code",
            1: "requirement",
            2: "validation",
            3: "management",
            4: "research" }
org_sal_band = {"sal-band-1": 1000000.0,
                "sal-band-2": 1500000.0,
                "sal-band-3":2000000.0}
```

Question-1

Which of the following code fragments will create a dictionary `org_role_dict` from the list `org_role_list`. [MSQ]

The output dictionary should be a equivalent to `org_role_list` and is given below.

Expected Output

```
{
    'dev': 'developer',
    'ba': 'business analyst',
    'qa': 'quality analyst',
    'lead': 'program manager',
    'ra': 'research associate'
}
```

(a)

```
org_role_dict = {}
for i in range(len(org_role_list)):
    org_role_dict.update({org_role_list[i][0] : org_role_list[i][1]})
print(org_role_dict)
```

(b)

```
org_role_dict = {}
for i in range(len(org_role_list)):
    org_role_dict.update([org_role_list[i]])
print(org_role_dict)
```

(c)

```
org_role_dict = {}
for i in range(len(org_role_list)):
    org_role_dict.update([(org_role_list[i][0], org_role_list[i][1])])
print(org_role_dict)
```

(d)

```
org_role_dict = {}
for unit, role in org_role_list:
    org_role_dict[unit] = role
print(org_role_dict)
```

Answer

(a), (b), (c), (d)

Solution

All the above options are correct. The function `update()` is used to add/update elements in a dictionary. Option (a) and (b) adds a `dict` of `key, value`. Option (b) adds a list of `key, value` tuple. Option (c) is another way to add/update am item in the dictionary.

Question-2

The list `emp_work` represents the work area of each employee on `emp_name` (given below). An employee at `i-th` position in `emp_name` is assigned the `(i%5)-th` item value of `org_work`. If the employee is at 4th position in `emp_name`, it is is assigned `4%5 = 4` which means `"research"` work area. Select the piece of code that that gives expected `emp_work`. [MSQ]

```
emp_name = ["Manohar", "Rajeev", "Savita", "Daizy", "Alok"]
```

Expected Output

```
['code', 'requirement', 'validation', 'management', 'research']
```

(a)

```
emp_work = []
for i in range(len(emp_name)):
    emp_work.append(org_work[i%5])
print(emp_work)
```

(b)

```
emp_work = []
for i in range(len(emp_name)):
    emp_work = emp_work + [org_work[i%5]]
print(emp_work)
```

(c)

```
emp_work = []
for i in range(emp_name):
    emp_work.append(org_work[i%5])
print(emp_work)
```

(d)

```
emp_work = []
for i in range(len(emp_name)):
    emp_work + [org_work[i%5]]
print(emp_work)
```

Answer

(a), (b)

Solution

Option (a) and (b) are correct. The function `append()` is used to add a new element to a list in the end. This function updates the original list itself and does not require return statement. Another way to concatenate two list is to use `+` operator. Option (c) is incorrect, it missed out the `len()` function inside the `range()` to get the length of the list. Option (d) requires `emp_work` to be

assigned the value of concatenate list at Line-3 to work. Hence, it is incorrect.

Question-3

Employee details and their salary bands are given below in the same order. The list `emp_sal` is holding annual salaries corresponding to the employees on the `emp_name`. Which of the following code fragment populates `emp_sal` correctly. [MSQ]

```
emp_name = ["Manohar", "Rajeev", "Savita", "Daizy", "Alok"]
emp_band = ["sal-band-1", "sal-band-2", "sal-band-1", "sal-band-3", "sal-band-2"]
```

(a)

```
emp_sal = []
for emp_id in range(len(emp_band)):
    emp_sal.append(org_sal_band[emp_band[emp_id]])
print(emp_sal)
```

(b)

```
emp_sal = []
for emp_id in range(len(emp_band)):
    emp_sal.append(org_sal_band.get(emp_band[emp_id]))
print(emp_sal)
```

(c)

```
emp_sal = []
for emp in emp_name:
    emp_sal.append(org_sal_band[emp_band[emp]])
print(emp_sal)
```

(d)

```
emp_sal = []
for emp_id in range(len(emp_name)):
    emp_sal.append(org_sal_band[emp_band[emp_id]])
print(emp_sal)
```

Answer

(a), (b), (d)

Solution

An element at a an index from a list can be retrieved using `list_var[index]` or `list_var.get(index)`. Option (a), (b) and (d) are correct. In option (c), the loop variable is a string representing element of `emp_name` rather than an `int`, it is not a valid list index. Therefore, (c) is incorrect.

Question-4

The salary of each employee in `emp_sal` (created in the last question) is hiked by say `x` percent due to good company performance. Which of the following code can be used to print the hiked salary of employee in the `emp_list`. The output should be employee name, old salary and new salary separated by comma. Each employee details is printed on a new line. [MSQ]

Example:

Old Salary	Hike %	New Salary
10000	10	10000 + 1000 = 11000

(a)

```
x = int(input())
for emp_id in range(len(emp_name)):
    print(emp_name[emp_id], emp_sal[emp_id], emp_sal[emp_id]*(x), sep=",")
```

(b)

```
x = int(input())
for emp_id in range(len(emp_name)):
    print(emp_name[emp_id], emp_sal[emp_id], emp_sal[emp_id]*(1+0.01*x),
sep=",")
```

(c)

```
x = int(input())
for emp_id in range(len(emp_name)):
    print(emp_name[emp_id], emp_sal[emp_id], emp_sal[emp_id]*(1+x), sep=",")
```

(d)

```
x = int(input())
for emp_id in range(len(emp_name)):
    print(emp_name[emp_id], emp_sal[emp_id], emp_sal[emp_id]*1.01*x, sep=",")
```

Answer

(b)

Solution

After hike, employee salary is `New Salary = Old Salary + Old Salary * x% = Old Salary* (1+0.01*x)`.

Option (b) performs the correct calculation of `New Salary`.

Problem-2

Questions 5 to 10 are based on a common theme

Use below code fragments to answer questions.

```
a = int(input())
b = int(input())
def v(a):
    return(a + w(b, a))

def w(a, b = 1):
    a += b
    return(a)

def x(a = 1, b = 1):
    a += b
    return(a)

def y(a = 1):
    global b
    a += b
    b += 1
    return(a)

def z(a):
    if a:
        a += z(a-1)
    return a
```

Question-5

Which of the following functions group uses default arguments partially or completely? [MCQ]

(a) `v, w, x`

(b) `v, w, z`

(c) `w, x, y`

(d) `x, y, z`

Answer

(c) `w, x, y`

Solution

By definition, function `w, x, y` uses default arguments. The **default** value is assigned by using the assignment (`=`) operator of the form `keyword_name = value` inside the parenthesis.

Question-6

Which of the following functions uses recursion? [MCQ]

(a) `v`

(b) `w`

(c) `x`

(d) `y`

(e) `z`

Answer

(e) `z`

Solution

A function is called `recursive` when it calls itself inside the function definition. Option (c) is the correct answer.

Question-7

What value function `z` returns when the argument is a positive integer `n`? [MCQ]

(a) `(n + 1)`

(b) `n(n + 1)/2`

(c) `2n`

(d) `n`

Answer

(b) `n(n + 1)/2`

Solution

For a positive integer argument, the function `z` recursively itself with a new argument which is one less than the value of last argument. `z(n)-> n+z(n) -> n+(n-1)+z(n-2) -> n+(n-1)+(n-2)+...+2+1+z(0)=n*(n+1)/2`, where `z(0) = 0`. This is a sum of a series of the first `n` natural numbers.

Question-8

What should be the expected output of the below code block, when `a = b = 10`. [MCQ]

```
print(v(a), a, b)
print(w(a), a, b)
print(x(a), a, b)
```

(a)

```
30 10 10
11 10 10
11 10 10
```

(b)

```
11 10 10
11 10 10
11 10 10
```

(c)

```
30 20 10
30 20 10
30 20 10
```

(d)

```
10 10 10
10 10 10
10 10 10
```

Answer

(a)

Solution

Both function `w` and `x` accepts two parameter `a` and `b` and returns sum of it. In this case for `a=10, b=10`, the output is `10`. Both uses `b` as default argument in the function call, which is 1. The function `x`, adds global variable to `a` to the output of `w (b, a)` which is 30. Hence, (a) is the correct answer.

Question-9

Given `a = 10, b = 20`, what should be the output of `print(y(a), a, b)`? [MCQ]

(a) `31 10 21`

(b) `30 11 21`

(c) `30 10 20`

(d) `30 10 21`

Answer

(d)

Solution

The function `y` accepts two parameter `a` and `b` and returns sum of it. Inside the function value of global variable `b` is also incremented by 1. For `a=10, b=20`, the output of print(y(a), a, b) will be `30 10 21`.

Question-10

Given `a = 10, b = 20`, what is the output of the following snippet of code?

```
y(a)
print(z(a), a, b)
```

(a) `50 10 21`

(b) `50 10 21`

(c) `51 11 21`

(d) `55 10 21`

Answer

(d)

Solution

The function `y` accepts two parameter `a` and `b` and returns sum of it. Inside the function value of global variable `b` is also incremented by 1. Function `z`, accepts a positive integer `n` as a parameter and returns sum of the first `n` natural numbers.

For `a=10, b=20`, the output of print(y(a), a, b) will be `55 10 21`.

Problem 3

Questions 11 to 13 are based on common theme

```
def doSomething(M, r, c):
    if len(M)*len(M[0]) == r * c:
        l = []
        for i in M:
            for j in i:
                l.append(j)
        M_ = []
        for i in range(r):
            M_.append([])
            for j in range(c):
                M_[-1].append(l.pop(0))
        return M_
```

Question-11

`doSomething` always returns a list for any given parameters.

(a) True
(b) False

Answer

(b) False

Solution

The return statement is defined inside the if-statement. Hence `doSomething` will return a list only when the `if` condition is satisfied.

Question-12

What will `doSomething(M, r, c)` return? `M` is a non-empty matrix (nested list), `r` and `c` are integers.

(a) Transpose of matrix `M`
(b) Reshaped matrix `M` of `r` rows and `c` columns for any integer `r` and `c`.
(c) Reshaped matrix `M` of `r` rows and `c` columns for specific `r` and `c`.
(d) Reshaped matrix `M` of `c` rows and `r` columns for specific `r` and `c`.

Answer

(c)

Solution

The function is will return a list only the if-condition `len(M)*len(M[0]) == r * c` is satisfied else `None` will be returned, hence it will return a list only when the product of `r` and `c` is equal to the product of number of rows `len(M)` and columns `len(M[0])`.

From the below code-snippet all the elements are made into a single list `l` with row wise ordered.

```
def doSomething(M, r, c):
    if len(M)*len(M[0]) == r * c:
        l = []
        for i in M:
            for j in i:
                l.append(j)
```

A new nested `M_` is created populated with the elements of list where the number of rows and columns will be `r` and `c` respectively.

```
        M_ = []
        for i in range(r):
            M_.append([])
            for j in range(c):
                M_[-1].append(l.pop(0))
        return M_
```

Question-13

For what values of `r` and `c` will `doSomething(M, r, c)` return a matrix? `len(M)` is equal to 12 and `len(M[0])` is 14. [MSQ]

(a) 24, 7
(b) 6, 28
(c) 2, 82
(d) 3, 56

Answer

(a), (b), (d)

Solution

The code results in the reshaping of the matrix. The number of elements before and after the operations should be equal. `len(M)` and `len(M[0])` gives number of rows and number of columns of the original matrix respectively. The total number of elements in the original matrix is `12*14=168`. Therefore, Option (a), (b), (d) are valid answers.

Problem-4

Questions 14 to 17 are based on common theme

Question-14

In the following code block, what should be the value of `a,b,c,d,e,f,g` to produce text given in the output.

```
x = "Running contests are of mostly three type {p}, {q} and {r}.".format(p = a, q = b, r = c)
y = f"Marathon an {a} sport, have two popular format {d} and {e}"
z = "The fastest human footspeed on record is %.2f km/h seen during a %d meter sprint by %s."%(g, h, f)
print(x, y, z, sep="\n")
```

Expected Output

```
Running contests are of mostly three type Endurance, Sprinting and Hurdling.
Marathon is an Endurance sport, there are two popular format Half Marathon and Full Marathon
The fastest human footspeed on record is 44.72 km/h seen during a 100 meter sprint by Usain Bolt.
```

Select the option that correctly matches items on Variable column to Value column and print above output. [MCQ]

Variable	Value
1. a	A. "Hurdling"
2. b	B. "Half Marathon"
3. c	C. 44.72
4. d	D. "Full Marathon"
5. e	E. 100
6. f	F. "Usain Bolt"
7. g	G. "Endurance"
8. h	H. "Sprinting"

(a) 1-G, 3-H, 2-A, 4-B, 6-D, 5-F, 7-C, 8-E

(b) 2-G, 1-H, 4-A, 3-B, 5-D, 6-F, 7-C, 8-E

(c) 3-G, 2-H, 1-A, 5-B, 4-D, 7-F, 6-C, 8-E

(d) 1-G, 2-H, 3-A, 4-B, 5-D, 6-F, 7-C, 8-E

Answer

(d) 1-G, 2-H, 3-A, 4-B, 5-D, 6-F, 7-C, 8-E

Solution

Below table shows the correct match of the options.

Variable	Value
1. a	G. "Endurance"
2. b	H. "Sprinting"
3. c	A. "Hurdling"
4. d	B. "Half Marathon"
5. e	D. "Full Marathon"
6. f	F. "Usain Bolt"
7. g	C. 44.72
8. h	E. 100

Question-15

Which of the following code will create a list `s` from three string variables `x, y, z` used in previous question. Hint: These variables represent a line in the below text in the same order. [MSQ]

```
Running contests are of mostly three type Endurance, Sprinting and Hurdling.
Marathon is an Endurance sport, there are two popular format Half Marathon and Full Marathon
The fastest human footspeed on record is 44.72 km/h seen during a 100 meter sprint by Usain Bolt.
```

(a)

```
s = [0]*3
s[0] = x
s[1] = y
s[2] = z
```

(b)

```
s = [0]
s[0] = x
s[1] = y
s[2] = z
```

(c)

```
s = [x, y, z]
```

(d)

```
s = [x] + [y] + [z]
```

Answer

(a), (c), (d)

Solution

A list can be concatenated using multiple ways. Options (a), (c), (d) are valid ways. Option (b) is not correct, because `s` is a list of only one element but we are trying to access 2nd and 3rd element. This should result in an error.

Question-16

The variable `s` used in the previous question holds a list of strings and is given below.

```
s = [
'Running contests are of mostly three type Endurance, Sprinting and Hurdling.',
'Marathon an Endurance sport, have two popular format Half Marathon and Full Marathon',
'The fastest human footspeed on record is 44.72 km/h seen during a 100 meter sprint by Usain Bolt.'
]
```

Which of the following options create a new nested list `t` given below (by splitting each element of `s` into a list of strings by space)? [MSQ]

```
[
['Running', 'contests', 'are', 'of', 'mostly', 'three', 'type', 'Endurance,', 'Sprinting', 'and', 'Hurdling.'],
['Marathon', 'an', 'Endurance', 'sport,', 'have', 'two', 'popular', 'format', 'Half', 'Marathon', 'and', 'Full', 'Marathon'],
['The', 'fastest', 'human', 'footspeed', 'on', 'record', 'is', '44.72', 'km/h', 'seen', 'during', 'a', '100', 'meter', 'sprint', 'by', 'Usain', 'Bolt.']
]
```

(a)

```
t = []
for i in [0,1,2]:
    t.append(s[i].split())
```

(b)

```
t = []
for i in range(3):
    t.append(s[i].split())
```

(c)

```
t = []
for i in range(len(s)):
    t.append(s[i].split(" ")) # a single space enclosed between double quotes
```

(d)

```
t = []
for q in s:
    t.append(q.split(" ")) # a single space enclosed between double quotes
```

Answer

(a), (b), (c), (d)

Solution

All options are valid. The string method `split()` breaks up a string at the specified separator and returns a list of string. By default the separator is a blank space string `' '`.

Question-17

The variable `t` used in the previous question holds a nested list of strings and is given below.

```
[
['Running', 'contests', 'are', 'of', 'mostly', 'three', 'type', 'Endurance,',
'Sprinting', 'and', 'Hurdling.'],
['Marathon', 'an', 'Endurance', 'sport,', 'have', 'two', 'popular', 'format',
'Half', 'Marathon', 'and', 'Full', 'Marathon'],
['The', 'fastest', 'human', 'footspeed', 'on', 'record', 'is', '44.72',
'km/h', 'seen', 'during', 'a', '100', 'meter', 'sprint', 'by', 'Usain',
'Bolt.']
]
```

```
print(t[0][a:b])
print(t[1][::c])
print(t[2][::-c])
```

The output of the code fragment is given below. [MCQ]

```
['Running', 'contests']
['Marathon', 'have', 'Half', 'Marathon']
['Bolt.', 'meter', 'seen', 'record', 'fastest']
```

What should be the value of `a, b, c` in order to get this output?

(a) `a = 0, b = 2, c = 4`

(b) `a = 1, b = 3, c = 4`

(c) `a = 0, b = 2, c = 5`

(d) `a = 0, b = 3, c = 4`

Answer

(a) `a = 0, b = 2, c = 4`

Solution

The first line is a slice (a sub list) of the first element of given nested list. The sub list contains only first two elements , hence `a=0, b=2` is acceptable.

The second line and third line of the output is printed when `c` is 4.

Hence, option (a) is the correct answer.