**MCQ**

1. What will be the output of the following code snippet?

**def func(a, b): return b if a == 0 else func(b % a, a) print(func(30, 75))**

* 1. **10**
  2. **20**
  3. **15**
  4. **0**

**Answer ---** The output of this code snippet is 15

1. **numbers = (4, 7, 19, 2, 89, 45, 72, 22) sorted\_numbers = sorted(numbers) even = lambda a: a % 2 == 0 even\_numbers = filter(even, sorted\_numbers) print(type(even\_numbers))**

* 1. **Int**
  2. **Filter**
  3. **List**
  4. **Tuple**

**Answer ---** filter

**3)** As what datatype are the \*args stored, when passed into a) Tuple

1. List
2. Dictionary
3. none

**Answer ---** tuple

1. **set1 = {14, 3, 55} set2 = {82, 49, 62} set3={99,22,17} print(len(set1 + set2 + set3))**

* 1. **105**
  2. **270**
  3. **0**
  4. **Error**

**Answer ---** error (type error)

1. What keyword is used in Python to raise exceptions? a) raise
2. try
3. goto
4. except

**Answer – raise**

**6)** Which of the following modules need to be imported to handle date time computations in Python?

1. timedate
2. date
3. datetime
4. time

**Answer--- datetime**

1. What will be the output of the following code snippet?

**print(4\*\*3 + (7 + 5)\*\*(1 + 1))**

* 1. **248**
  2. **169**
  3. **208**
  4. **233**

**Answer --- 208**

1. Which of the following functions converts date to corresponding time in Python? a) strptime
2. strftime
3. both a) and b)
4. None

**Answer --- none**

**9)** The python tuple is \_\_\_\_\_ in nature. a) mutable

b)immutable

c)unchangeable

d) none

**Answer --- immutable**

10)

The \_\_\_ is a built-in function that returns a range object that consists series of integer numbers, which we can iterate using a for loop.

1. range()
2. set()
3. dictionary{}
4. None of the mentioned above

**Answer --- range()**

**Question 11**

**Amongst which of the following is a function which does not have any name?**

1. Del function
2. Show function
3. Lambda function
4. None of the mentioned above

**Answer --- lamba function**

Question 12

**The module Pickle is used to \_\_\_.**

1. Serializing Python object structure
2. De-serializing Python object structure
3. Both A and B
4. None of the mentioned above

**Answer --- Both A and B**

Question 13

**Amongst which of the following is / are the method of convert Python objects for writing data in a binary file?**

1. set() method
2. dump() method
3. load() method
4. None of the mentioned above

**Answer --- dump() method**

14

**Amongst which of the following is / are the method used to unpickling data from a binary file?**

1. load()
2. set() method
3. dump() method
4. None of the mentioned above

**Answer --- load()**

15.

**A text file contains only textual information consisting of \_\_\_.**

1. Alphabets
2. Numbers
3. Special symbols
4. All of the mentioned above

**Answer --- All of above**

16

Which Python code could replace the ellipsis (...) below to get the following output? (Select all that apply.) captains = {

"Enterprise": "Picard",

"Voyager": "Janeway",

"Defiant": "Sisko", }

Enterprise Picard,

Voyager Janeway

Defiant Sisko

1. for ship, captain in captains.items():

print(ship, captain)

1. for ship in captains:

print(ship, captains[ship])

1. for ship in captains:

print(ship, captains)

1. both a and b

**Answer --- bothe A and B**

**17)**

|  |
| --- |
| captains |

Which of the following lines of code will create an empty dictionary named ? a) captains = {dict}

1. type(captains)
2. captains.dict()
3. captains = {}

**Answer --- captains \_{dict}**

**18)** Now you have your empty dictionary named captains. It’s time to add some data!

Specifically, you want to add the key-value pairs "Enterprise": "Picard", "Voyager": "Janeway", and "Defiant": "Sisko".

Which of the following code snippets will successfully add these key-value pairs to the existing captains dictionary?

1. captains{"Enterprise" = "Picard"} captains{"Voyager" = "Janeway"} captains{"Defiant" = "Sisko"}

1. captains["Enterprise"] = "Picard" captains["Voyager"] = "Janeway" captains["Defiant"] = "Sisko"

1. captains = {

"Enterprise": "Picard",

"Voyager": "Janeway",

"Defiant": "Sisko",

}

1. None of the above

**Answer ---- b) captains["Enterprise"] = "Picard" captains["Voyager"] = "Janeway" captains["Defiant"] = "Sisko"**

1. **)** You’re really building out the Federation Starfleet now! Here’s what you have: captains = {

"Enterprise": "Picard",

"Voyager": "Janeway",

"Defiant": "Sisko",

"Discovery": "unknown",

}Now, say you want to display the ship and captain names contained in the dictionary, but you also want to provide some additional context. How could you do it?

* 1. for item in captains.items():

print(f"The [ship] is captained by [captain].")

* 1. for ship, captain in captains.items():

print(f"The {ship} is captained by {captain}.")

* 1. for captain, ship in captains.items():

print(f"The {ship} is captained by {captain}.")

* 1. All are correct

**Answer ---** **b) for ship, captain in captains.items():**

**print(f"The {ship} is captained by {captain}.")**

1. **)**

You’ve created a dictionary, added data, checked for the existence of keys, and iterated over it with a for loop. Now you’re ready to delete a key from this dictionary:

captains = {

"Enterprise": "Picard",

"Voyager": "Janeway",

"Defiant": "Sisko",

"Discovery": "unknown",

}

|  |
| --- |
| "Discovery" |

What statement will remove the entry for the key ?

* 1. del captains
  2. captains.remove()
  3. del captains["Discovery"]
  4. captains["Discovery"].pop()

**Answer --- c) del captains["Discovery"]**