

Python Code

```
1  import random
2  """
3  TUPLE QUESTIONS
4
5  t1 = (10,20,"Python","Code")
6  t2=("A","B")
7  1. Access and print the first element of t1.
8  2. Access and print the last element of t2.
9  3. Convert t1 into a list, change "Code" to "Program", and convert it back into a tuple.
10 4. Unpack all elements of t1 into separate variables and print them.
11 5. Join t1 and t2 and store the result in a new tuple.
12 6. Access elements from index 1 to 3 of the joined tuple.
13 t1=(1,2,3)
14 7. Multiply the tuple t1 by 3 and print the result.
15 """
16 t1 = (10,20,"Python","Code")
17 t2=("A","B")
18
19 xlist=[
20     t1.__getitem__(0),
21     t2.__getitem__(-1),
22     (ylist := list(t1), ylist.__setitem__(3, "Program"), tuple(ylist))[2],
23     [f"{a}|{b}|{c}|{d}" for a,b,c,d in [t1]][0],
24     t3:= (t1+t2),
25     t3[1:4],
26     t1*3
27 ]
28
29 for i,x in enumerate(xlist):
30     print(f"{i+1}). {x}")
31
32 print("=====")
33 #####
34 """
35 SET QUESTIONS
36 mySet={10,20,30,40}
37 1. Print the given set.
38 2. Check whether 20 is present in my_set.
39 3. Find and print the length of my_set.
40 4. Add 50 to my_set.
41 5. Remove 30 from my_set.
42 6. Remove an element using discard() and note the output.
43 7. Remove a random element from my_set.
44 8. Loop through my_set and print each element.
45 9. Clear all elements from my_set.
46 """
47
48 mySet={10,20,30,40}
```

```

49 xlist=[
50     mySet.copy(),
51     f'{"Yes, 20 is present in mySet." if mySet.__contains__(20) else "No, 20 is not present in mySet."}',
52     mySet.__len__(),
53     (mySet.add(50), mySet.copy())[1],
54     (mySet.remove(30), mySet.copy())[1],
55     (mySet.discard(50), mySet.copy())[1],
56     (mySet.remove(random.choice(list(mySet))), mySet.copy())[1],
57     [f"{i}" for i in mySet],
58     mySet.clear() or mySet.copy(),
59 ]
60
61 for i, x in enumerate(xlist):
62     print(f"{i+1}). {x}")
63
64 print("=====")
65 #####
66
67 """
68 set1={1,2,3}
69 set2={3,4,5,6}
70 1. Print the elements that are in set1 but not in set2.
71 2. Print the elements that are in either set1 or set2 but not in both (symmetric difference).
72 3. Add all items from set2 into set1 and print the updated set1.
73 4. Print all unique elements from both set1 and set2.
74 5. Add 7 to set1 and print the set.
75 """
76
77 set1={1,2,3}
78 set2={3,4,5,6}
79
80 xlist=[
81     (set1 - set2),
82     (set1 ^ set2),
83     (set1.update(set2), set1.copy())[1],
84     (set1 | set2),
85     set1.add(7) or set1.copy()
86 ]
87
88 for i,x in enumerate(xlist):
89     print(f"{i+1}). {x}")
90
91 print("=====")
92 #####
93
94 """
95 DICTIONARY QUESTIONS
96 student{
97     "name": "Anu",
98     "age": 20,
99     "course": "Python"

```

```

100 }
101
102 1. Print all keys, values, and all items in the dictionary.
103 2. Access the value of "name".
104 3. Access the value of "course" using get().
105 4. Add a new key "marks" with value 85.
106 5. Update the value of "age" to 21.
107 6. Remove "course" using pop().
108 7. Remove the last inserted item using popitem().
109 8. Loop through the dictionary and print keys and values.
110 """
111
112 student={
113     "name":"Anu",
114     "age":20,
115     "course":"Python"
116 }
117 xlist=[
118     (list(student.keys()),list(student.values()),list(student.items())),
119     student["name"],
120     student.get("course"),
121     (student.update({"marks":85}), student.copy())[1],
122     (student.update({"age":21}), student.copy())[1],
123     (student.pop("course"), student.copy())[1],
124     (student.popitem(), student.copy())[1],
125     [f"{k}: {v}" for k,v in student.items()]
126 ]
127
128 for i,x in enumerate(xlist):
129     print(f"{i+1}). {x}")
130 print("=====")
131 #####
132
133 """
134 students{
135     "student1":{
136         "name":"Anu",
137         "age":20
138     },
139     "student2":{
140         "name":"Ravi",
141         "age":21
142     }
143 }
144
145
146 1. Print the complete nested dictionary.
147 2. Create a copy of the students dictionary.
148 """
149
150 students={
151     "student1":{

```

```

152         "name": "Anu",
153         "age": 20
154     },
155     },
156     "student2": {
157         "name": "Ravi",
158         "age": 21
159     }
160 }
161 xlist = [
162     f"{students}",
163     students.copy()
164 ]
165
166 for i, x in enumerate(xlist):
167     print(f"{i+1}). {x}")
168
169 print("=====")
170 #####
171
172 """
173 employee = {
174     "emp_id": 101,
175     "name": "Kiran",
176     "department": "HR",
177     "salary": 35000
178 }
179
180 1. Display all keys in the employee dictionary.
181 2. Display all values in the employee dictionary.
182 3. Display all key-value pairs (items) from the dictionary.
183 4. Access and print the value of "name".
184 5. Update the "salary" to 40000.
185 """
186
187 employee = {
188     "emp_id": 101,
189     "name": "Kiran",
190     "department": "HR",
191     "salary": 35000
192 }
193
194 xlist = [
195     list(employee.keys()),
196     list(employee.values()),
197     list(employee.items()),
198     employee["name"],
199     (employee.update({"salary": 40000}), employee.copy())[1]
200 ]
201
202 for i, x in enumerate(xlist):
203     print(f"{i+1}). {x}")

```

```
1). 10
2). B
3). (10, 20, 'Python', 'Program')
4). 10|20|Python|Code
5). (10, 20, 'Python', 'Code', 'A', 'B')
6). (20, 'Python', 'Code')
7). (10, 20, 'Python', 'Code', 10, 20, 'Python', 'Code', 10, 20, 'Python', 'Code')
=====
1). {40, 10, 20, 30}
2). Yes, 20 is present in mySet.
3). 4
4). {50, 20, 40, 10, 30}
5). {40, 10, 50, 20}
6). {40, 10, 20}
7). {10, 20}
8). ['10', '20']
9). set()
=====
1). {1, 2}
2). {1, 2, 4, 5, 6}
3). {1, 2, 3, 4, 5, 6}
4). {1, 2, 3, 4, 5, 6}
5). {1, 2, 3, 4, 5, 6, 7}
=====
1). ([('name', 'age', 'course'], ['Anu', 20, 'Python'], [(('name', 'Anu'), ('age', 20), ('course', 'Python'))])
2). Anu
3). Python
4). {'name': 'Anu', 'age': 20, 'course': 'Python', 'marks': 85}
5). {'name': 'Anu', 'age': 21, 'course': 'Python', 'marks': 85}
6). {'name': 'Anu', 'age': 21, 'marks': 85}
7). {'name': 'Anu', 'age': 21}
8). ['name: Anu', 'age: 21']
=====
1). {'student1': {'name': 'Anu', 'age': 20}, 'student2': {'name': 'Ravi', 'age': 21}}
2). {'student1': {'name': 'Anu', 'age': 20}, 'student2': {'name': 'Ravi', 'age': 21}}
=====
1). ['emp_id', 'name', 'department', 'salary']
2). [101, 'Kiran', 'HR', 35000]
3). [('emp_id', 101), ('name', 'Kiran'), ('department', 'HR'), ('salary', 35000)]
4). Kiran
5). {'emp_id': 101, 'name': 'Kiran', 'department': 'HR', 'salary': 40000}
```