NAME **ADNAN**

ID **13507**

SUBJECT **MODERN PROGRAMMING LANGUAGES…**

Q1.

Solved.

Restaurant Example:

Class Restaurant ():

“”” Restaurant class Representing a Restaurant.”””””

def _ init _ (self,name , cuisine _ type):

self.name= name. title ()

self. cuisine _type = cuisine _ type ()

def describe _ restaurant (self):

“”” Show the summary of the restaurant.”””

msg = self. name + “serves wonderful + “ self. cuisine _ type + “.”

print (“\n” + msg)

def open _ restaurant (self):

“”” Restaurant is open message display.”””

msg = self. name + **“ is open . come on in! “**

print (“\n” + msg)

restaurant = Restaurant (‘ **Basic Kneads Pizza’**)

print (restaurant.name)

------ continuous-----

print (restaurant. cuisine _type)

restaurant. describe _ restaurant ()

restaurant. open _ restaurant ()

Output:

Basic Knead Pizza

Basic Knead Pizza Serves Wonderful.

Basic Knead Pizza is open. Come on in!

------- -------- ------- --------- -------- ------- ------ -------- ------ --------- ------ ------ -------- ----

Q2.

Solved.

def make _ album(artist ,title):

“”” Information containing about an album.”””

Album _ dict = {

‘artist’: artist .title (),

‘title’ : title . title (),

}

Return album_dict

album = make _ album (‘Michael Jackson’, ‘Thriller’)

print (album)

album = make _ album (‘Meat Loaf’, ‘Bat Out of Hell’)

print (album)

album = make _ album (‘Shania Twain’, ‘Come on Over’)

print (album)

------ continuous-----

Output:

{ 'title': 'Thriller' , 'artist' : 'Michael Jackson' }

{ 'title': 'Bat Out of Hell', 'artist' : 'Meat Loaf'}

{ 'title': 'Come on Over', 'artist' 'Shania Twain'}

Q3.

(a)

Solved.

```
car = input ("What kind of car would you like?")
print ("Let me see if I can find you a " + car .title()+ ".")
```

Output:

What kind of car would you like?Corolla Toyota car

Let me see if I can find you a Corolla Toyota car.

(b)

Solved.

```
Party _ size = input ("How many people are in your dinner party tonight?")
Party_ size = int (party _ size)
If party _ size >8:
Print ("' I am sorry, you ' ll wait for a table.")
else:
print ("your table is ready.")
```

-----continuous----

Output:

How many people are in your dinner party tonight?16

I am sorry, you ‘ ll wait for a table.

Or

How many people are in your dinner party tonight?4

Your table is ready.

------- -------- ------- --------- -------- ------- ------ -------- ------ --------- ------ ------ -------- ----

(C)

Solved.

```
number = input (“Give me a number, please:”)
number = int ( number)
if number % 10 == 0:
print (str(number) +” is a multiple of 10.”)
else:
print (str(number) +” is not a multiple of 10.”)
```

Output:

Give me a number, please:26

26 is not a multiple of 10.

Or

Give me a number, please:60

60 is a multiple of 10.

------- -------- ------- --------- -------- ------- ------ -------- ------ --------- ------ ------ -------- ----

(d)

Solved.

```
number = input (“Give me a number, please:”)
number = int ( number)
if number % 10 == 0:
print (str(number) +” is a multiple of 20.”)
else:
print (str(number) +” is not a multiple of 20.”)
```

Output:

Give me a number, please:33

33 is not a multiple of 20.

Or

Give me a number, please:100

100 is a multiple of 20.

------- -------- ------- --------- -------- ------- ------ -------- ------ --------- ------ ------ -------- ----

(e)

Solved.

```
number = input (“Give me a number, please:”)
number = int (number)
if number % 10== 0:
print (str(number) +” is a multiple of 30.”)
else:print (str(number) +” is not a multiple of 30.”)
```

Output:

Give me a number, please:56

56 is not a multiple of 30.

Or

Give me a number, please:90

90 is a multiple of 30.

(f)

Solved.

```
number = input (“Give me a number, please:”)
number = int ( number)
if number % 10 == 0:
print (str(number) +” is a multiple of 140.”)
else:
print (str(number) +” is not a multiple of 140.”)
```

Output:

Give me a number, please:180

180 is not a multiple of 140.

Or

Give me a number, please:280

280 is a multiple of 140.

Q4.

Solved.

```
Prompt =  “\n What topping you like on your pizza?”
Prompt + =”\n Enter ‘quit’ when you are finished: “
```

While True:

Topping = input(prompt)

If topping ! = ‘quit’:

Print (“ I’ ll add “ + topping + “ to your pizza. “)

While reverse:

Topping. reverse = input (prompt.reverse)

If topping ! = ‘quit’:

Print (“ I ‘ ll add “ + topping + “ reverse to your pizza.”)

else:

break

Output:

What topping you like on your pizza?

Enter ‘quit’ when you are finished:sausage

I’ ll add sausage to your pizza.

What topping you like on your pizza?

Enter ‘quit’ when you are finished:pepperoni

I’ ll add pepperoni to your pizza.

What topping you like on your pizza?

Enter ‘quit’ when you are finished:bacon

I’ ll add bacon to your pizza.

What topping you like on your pizza?

Enter ‘quit’ when you are finished: quit

Sort order

What topping you like on your pizza?

Enter ‘quit’ when you are finished: sausage,pepperoni,bacon

I’ ll add sausage,pepperoni,bacon to your pizza.

<u>Print in reverse order</u>

What topping you like on your pizza?

Enter ‘quit’ when you are finished: quit

What topping you like on your pizza?

Enter ‘quit’ when you are finished:bacon

I’ ll add bacon to your pizza.

What topping you like on your pizza?

Enter ‘quit’ when you are finished:pepperoni

I’ ll add pepperoni to your pizza.

What topping you like on your pizza?

Enter ‘quit’ when you are finished:sausage

I’ ll add sausage to your pizza.

------- -------- ------- --------- -------- ------- ------ -------- ------ --------- ------ ------ -------- ----

Q5.

Solved.

```
rivers = {
rivers              country
'Fraser' :          'canada',
'nile':         'Egypt',
'yangtaze' : 'china',

}
""" country 1 , river 1"""
 For river, country in river ():
Print (f "The{ river.title()}  Flows through { country.title()}.")
""" country 2 , river 2"""
For river, country in river ():
Print (f "The{ river.title()}  Flows through { country.title()}.")
""" country 3 , river 3"""
For river, country in river ():
Print (f "The{ river.title()}  Flows through { country.title()}.")
Print ("\n The following rivers are included in this data set:")
for river in rivers.keys ():
print (f "- { river,title()}"}

Print ("\n The following countries are included in this data set:")
for country in rivers. value ():
print (f "- { country.title()}"}
```

Output:

The Fraser Flows through to Canada.

The nile Flows through to Egypt.

The yangtaze Flows through to china.

The following rivers are included in this data set:

Fraser

Nile

Yangtaze

The following countries are included in this data set:

Canada

Egypt

China

------- -------- ------- --------- -------- ------- ------ -------- ------ --------- ------ ------

---------- THE END--------