

Show Attended Group? :	No
Edit Attended Group? :	No
Break time :	0
Group Marks :	1090
Is this Group for Examiner? :	No
Examiner permission :	Cant View
Show Progress Bar? :	No
Revisit allowed for group Instructions? :	Yes
Maximum Instruction Time :	0
Minimum Instruction Time :	0
Group Time In :	Minutes
Navigate To Group Summary From Last Question? :	No
Disable Submit Button During Assessment? :	No
Section Selection Time? :	0
No of Optional sections to be attempted :	0

Sem2 Intro to python

Section Id :	64065364114
Section Number :	1
Section type :	Online
Mandatory or Optional :	Mandatory
Number of Questions :	18
Number of Questions to be attempted :	18
Section Marks :	50
Display Number Panel :	Yes
Section Negative Marks :	0
Group All Questions :	No
Enable Mark as Answered Mark for Review and Clear Response :	No
Maximum Instruction Time :	0
Sub-Section Number :	1
Sub-Section Id :	640653133979
Question Shuffling Allowed :	No

Question Number : 1 Question Id : 640653903542 Question Type : MCQ Calculator : Yes
Correct Marks : 0

Question Label : Multiple Choice Question

**THIS IS QUESTION PAPER FOR THE SUBJECT "FOUNDATION LEVEL : SEMESTER II:
INTRODUCTION TO PYTHON (COMPUTER BASED EXAM)"**

ARE YOU SURE YOU HAVE TO WRITE EXAM FOR THIS SUBJECT?

CROSS CHECK YOUR HALL TICKET TO CONFIRM THE SUBJECTS TO BE WRITTEN.

(IF IT IS NOT THE CORRECT SUBJECT, PLS CHECK THE SECTION AT THE TOP FOR THE SUBJECTS REGISTERED BY YOU)

Options :

6406533042671. ✓ YES

6406533042672. ✗ NO

Question Number : 2 Question Id : 640653903543 Question Type : MCQ Calculator : Yes

Correct Marks : 0

Question Label : Multiple Choice Question

Useful Data

Presentation

There are two types of blocks that you would see in all the questions:

Code

```
for i in range(10):  
    if i % 2 == 0:  
        print(i)
```

Input or Output

0
2
4
6
8

In both the blocks, please note that the region to the left of the thin vertical line — | — corresponds to line-numbers. Do not confuse the line numbers with the content of the code or the input-output. Just to be clear:

Line Numbers ← → Code/Input/Output

1	0
2	2
3	4
4	6
5	8

Useful information

range

Sample behaviour of the `range` function:

- `range(5)` corresponds to the sequence 0, 1, 2, 3, 4
- `range(1, 5)` corresponds to the sequence 1, 2, 3, 4
- `range(1, 1)` is the empty sequence

// operator

// is the floor division operator. `5 // 2` is 2 and *not* 2.5

NAT → integer

For all NAT questions in this exam, the answer will always be an integer and not a float value.

If the answer to a question is 18, then just enter that value. Do *not* enter 18.0

Options :

6406533042673. ✓ Useful Data has been mentioned above.

6406533042674. ✗ This data attachment is just for a reference & not for an evaluation.

Sub-Section Number :

2

Sub-Section Id :

640653133980

Question Shuffling Allowed :

Yes

Question Number : 3 Question Id : 640653903544 Question Type : MCQ Calculator : Yes

Correct Marks : 3

Question Label : Multiple Choice Question

Which of the following code blocks prints the sum of the digits for the given number?

Options :

```
n = input()
total = 0
for digit in n:
    total = total + digit
print(total)
```

6406533042675. ✖

```
n = input()
total = 1
for digit in n:
    total = total + int(digit)
print(total)
```

6406533042676. ✖

```
n = input()
total = 0
for digit in n:
    total = total + int(digit)
print(total)
```

6406533042677. ✔

```
n = input()
for digit in n:
    total = total + int(digit)
print(total)
```

6406533042678. ✖

Question Number : 4 Question Id : 640653903545 Question Type : MCQ Calculator : Yes

Correct Marks : 3

Question Label : Multiple Choice Question

What is the output of the following snippet of code?

```
def calculate_area(length, width):  
    try:  
        area = length * width  
        return area  
    except TypeError:  
        print('Both length and width must be numbers')  
    finally:  
        print('Reached here')  
  
a = calculate_area('10', '5')  
b = calculate_area(10, None)  
c = calculate_area(10, 5)  
print(c)
```

Options :

Both length and width must be numbers
50

6406533042679. ✖

50

6406533042680. ✖

Reached here
Reached here
Reached here
50

6406533042681. ✖

Both length and width must be numbers
Reached here
Both length and width must be numbers
Reached here
Reached here
50

6406533042682. ✔

Question Number : 5 Question Id : 640653903546 Question Type : MCQ Calculator : Yes

Correct Marks : 3

Question Label : Multiple Choice Question

`check_temperature` is a method of the class `Oven` that updates the attribute `preheat_required` to the Boolean literal `True` if the attribute `temperature` is less than 180 degrees Celsius, indicating that the oven needs preheating, and `False` otherwise. Select the correct implementation of this method.

Options :

```
def check_temperature(self):  
    if self.temperature < 180:  
        self.preheat_required = True  
    else:  
        self.preheat_required = False
```

6406533042683. ✓

```
def check_temperature():  
    if self.temperature < 180:  
        self.preheat_required = True  
    else:  
        self.preheat_required = False
```

6406533042684. ✗

```
def check_temperature(self):  
    if temperature < 180:  
        self.preheat_required = True  
    else:  
        self.preheat_required = False
```

6406533042685. ✗

```
def check_temperature(self):  
    if self.temperature <= 180:  
        preheat_required = True  
    else:  
        preheat_required = False
```

6406533042686. ✗

Question Number : 6 Question Id : 640653903547 Question Type : MCQ Calculator : Yes

Correct Marks : 3

Question Label : Multiple Choice Question

What will be the output of the code below.

```
nums = [1,2,1]
value = 0

for i in range(3):
    for x in range(len(nums)):
        value = value + x
    for y in nums:
        value = value - y

print(value)
```

Options :

6406533042687. ✖ 12

6406533042688. ✖ 9

6406533042689. ✖ 0

6406533042690. ✔ -3

Question Number : 7 Question Id : 640653903548 Question Type : MCQ Calculator : Yes

Correct Marks : 3

Question Label : Multiple Choice Question

What will be the output of the code below.

```
def matrix_operation(matrix):  
    for j in range(len(matrix[0])):  
        for i in range(len(matrix)):  
            if i == 0 or i == len(matrix) - 1:  
                matrix[i][j] = 1  
  
    for row in matrix:  
        print(row)  
  
A = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]  
matrix_operation(A)
```

Options :

```
[1, 1, 1]  
[1, 5, 1]  
[1, 1, 1]
```

6406533042691. ✖

```
[1, 1, 1]  
[4, 5, 6]  
[1, 1, 1]
```

6406533042692. ✔

```
[1, 2, 1]  
[1, 5, 1]  
[1, 8, 1]
```

6406533042693. ✖

```
[1, 1, 1]  
[1, 1, 1]  
[1, 1, 1]
```

6406533042694. ✖

Question Label : Multiple Choice Question

Consider the file `encrypted.txt` given below, also consider the python file both in the same directory. What will be the output of the code below?

`encrypted.txt`

```
psgiy  
ryiken  
ontzn  
mtlhzn  
paaho  
txbinn
```

```
file = open('encrypted.txt', 'r')  
  
word = ""  
for i in range(6):  
    line = file.readline()  
    word += line[i]  
  
print(word)  
  
file.close()
```

Options :

6406533042695. ✓ `python`

6406533042696. ✗ `prompt`

6406533042697. ✗ `gitlab`

6406533042698. ✗ `syntax`

Sub-Section Number :

3

Sub-Section Id :

640653133981

Question Shuffling Allowed :

Yes

Question Number : 9 Question Id : 640653903555 Question Type : MSQ Calculator : Yes

Correct Marks : 4 Max. Selectable Options : 0

Question Label : Multiple Select Question

Which of the following code blocks correctly implements to classify an integer as "Positive", "Negative", or "Zero"?

Options :

```
def classify_number(num):  
    if num >= 0:  
        return "Positive"  
    elif num <= 0:  
        return "Negative"  
    else:  
        return "Zero"
```

6406533042704. ✖

```
def classify_number(num):  
    if num == 0:  
        return "Zero"  
    elif num > 0:  
        return "Positive"  
    else:  
        return "Negative"
```

6406533042705. ✔

```
def classify_number(num):  
    if num > 0:  
        return "Positive"  
    elif num < 0:  
        return "Negative"  
    else:  
        return "Zero"
```

6406533042706. ✔

6406533042707. ✖

```
def classify_number(num):  
    if num < 0:  
        return "Negative"  
    elif num > 0:  
        return "Zero"  
    else:  
        return "Positive"
```

Sub-Section Number :

4

Sub-Section Id :

640653133982

Question Shuffling Allowed :

Yes

Question Number : 10 Question Id : 640653903556 Question Type : MSQ Calculator : Yes

Correct Marks : 3 Max. Selectable Options : 0

Question Label : Multiple Select Question

The following snippets are supposed to do the following: if n is a positive integer, then print the sum of the first $2n$ positive integers.

Snippet-1

```
P = [x for x in range(1, 2 * n + 1)]  
Q = [P[i] for i in range(0, n)]  
print(sum(Q))
```

Snippet-2

```
P = [x for x in range(1, 2 * n + 1)]  
Q = [P[i] + P[-i - 1] for i in range(0, n)]  
print(sum(Q))
```

Choose all the correct options regarding the given code .

Options :

6406533042708. ✖ Snippet-1 is correct

6406533042709. ✔ Snippet-2 is correct

6406533042710. ✔

If `n` is 3, the **incorrect** snippet generates the output :

6

If `n` is 3, the **incorrect** snippet generates the output :

12

6406533042711. ✖

Question Number : 11 Question Id : 640653903557 Question Type : MSQ Calculator : Yes

Correct Marks : 3 Max. Selectable Options : 0

Question Label : Multiple Select Question

Consider the following python code snippet.

```
collection1 = {2, 3, 5, 7, 9, 3}
collection2 = (7, 3, 5, 7)
```

CODE-BLOCK

```
print(collection1)
print(collection2)
```

Select all the options which will not raise an exception when filled in place of `CODE-BLOCK` .

Options :

```
collection1.add(3)
collection1 = collection1 & set(collection2)
```

6406533042712. ✔

```
collection1.union({4})
collection2[3] = 8
```

6406533042713. ✖

```
collection1.add([4,6])
collection2 = list(collection2)
```

6406533042714. ✖

6406533042715. ✔

```
collection1.add(collection2)
collection2 = list(collection2)
```

Sub-Section Number :

5

Sub-Section Id :

640653133983

Question Shuffling Allowed :

Yes

Question Number : 12 Question Id : 640653903550 Question Type : SA Calculator : None

Correct Marks : 3

Question Label : Short Answer Question

Consider the Python code given below.

```
f = open('output.txt', 'w')
f.write('arjun\n')
f.write('sreethu')
f.write('''jinto
sijo
''')
f.close()
```

How many number of lines will be present in the file `output.txt` after the execution of the above code?

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

3

Question Number : 13 Question Id : 640653903551 Question Type : SA Calculator : None

Correct Marks : 3

Question Label : Short Answer Question

What will be the output of the following code.

```
a = [3, 4, 5, 6]
i = -2
while i < 0:
    print(a[i]*i)
    i += 2
```

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

-10

Question Number : 14 **Question Id :** 640653903552 **Question Type :** SA **Calculator :** None

Correct Marks : 3

Question Label : Short Answer Question

What will be the output of the code below.

```
numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
result = []

for num in numbers:
    if num % 2 == 0:
        result.append(num // 2)

result.reverse()
result.extend([11, 12, 13])
result.remove(5)

print(result[4])
```

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

11

Question Number : 15 Question Id : 640653903553 Question Type : SA Calculator : None

Correct Marks : 3

Question Label : Short Answer Question

What will be the output of the python code below?

```
data = {  
    "A": 5,  
    "B": 10,  
    "C": 15,  
    "D": 20  
}  
  
for key in list(data.keys()):  
    if data[key] % 2 == 0:  
        data[key] = data[key] * 5  
    if data[key] % 10 == 0:  
        data[key] = data[key] // 2  
    else:  
        del data[key]  
  
sum = 0  
for key, value in data.items():  
    sum += value  
  
print(sum)
```

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

75

Question Number : 16 Question Id : 640653903554 Question Type : SA Calculator : None

Correct Marks : 3

Question Label : Short Answer Question

Consider the following snippet of code.

```
def recursive_func(n):  
    if n == 0:  
        return 10  
    return n + recursive_func(n - 2)  
  
result = recursive_func(10)  
print(result)
```

What will be the output of the code?

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

40

Sub-Section Number :

6

Sub-Section Id :

640653133984

Question Shuffling Allowed :

No

Question Id : 640653903558 **Question Type :** COMPREHENSION **Sub Question Shuffling Allowed :** No **Group Comprehension Questions :** No **Question Pattern Type :** NonMatrix **Calculator :** None

Question Numbers : (17 to 18)

Question Label : Comprehension

Consider the class `Student` :

```
class Student:
    def __init__(self, name, age, marks):
        self.name = name
        self.age = age
        self.marks = marks

    def display_details(self):
        print(self.name)
        print(self.age)
        print(self.marks)

    def is_passing(self):
        passing_marks = 40
        return self.marks >= passing_marks
    def extra_curricular_points(self):
        return 5
```

`Topper` is a sub-class of `Student`:

```
class Topper(Student):
    def __init__(self, name, age, marks, rank):
        super().__init__(name, age, marks)
        self.rank = rank

    def display_details(self):
        # CODE-BLOCK

    def is_topper(self):
        return self.rank == 1

    def extra_curricular_points(self):
        if self.marks > 90:
            return 20
        else:
            return 15

student1 = Topper("Arun", 21, 95, 1)
student1.display_details()
```

Based on the above data, answer the given subquestions.

Sub questions

Question Number : 17 Question Id : 640653903559 Question Type : MSQ Calculator : Yes

Correct Marks : 1.5 Max. Selectable Options : 0

Question Label : Multiple Select Question

What should be filled in place of `CODE-BLOCK` so that it generates the following output?

```
Arun
21
95
1
```

Options :

```
super.display_details()
print(self.rank)
```

6406533042716. ✖

```
Student.display_details()
print(self.rank)
```

6406533042717. ✖

```
super().display_details()
print(self.rank)
```

6406533042718. ✔

```
Student.display_details(self)
print(self.rank)
```

6406533042719. ✔

Question Number : 18 Question Id : 640653903560 Question Type : MCQ Calculator : Yes Correct Marks : 1.5

Question Label : Multiple Choice Question

What will be the output if the following line is executed?

```
print(student1.extra_curricular_points())
```

Options :

6406533042720. ✖ This will generate an error.

6406533042721. ✖

6406533042722. ✖

6406533042723. ✔ 20

Sub-Section Number :

7

Sub-Section Id :

640653133985

Question Shuffling Allowed :

No

Question Id : 640653903561 Question Type : COMPREHENSION Sub Question Shuffling Allowed : No Group Comprehension Questions : No Question Pattern Type : NonMatrix Calculator : None

Question Numbers : (19 to 20)

Question Label : Comprehension

Execute the following snippet of code and answer the given sub-questions

```
data = [  
    {"type": "regular", "quantity": 2, "price": 20},  
    {"type": "premium", "quantity": 1, "price": 15},  
]  
  
for order in data:  
    if order["type"] == "premium":  
        rate = 0.1  
    else:  
        rate = 0  
    order["total"] = (1 - rate) * (order["price"]) * order["quantity"]  
  
for order in data:  
    print(f"Type: {order['type']}, Total: {order['total']:.2f}")
```

Sub questions

Question Number : 19 Question Id : 640653903562 Question Type : MCQ Calculator : Yes Correct Marks : 2

Question Label : Multiple Choice Question

What will be printed when the code is run?

Options :

```
Type: 2, Total: 50.00  
Type: 1, Total: 15.50
```

6406533042724. ✖

```
Type: regular, Total: 40.00  
Type: premium, Total: 13.50
```

6406533042725. ✔

```
Type: regular, Total: 40  
Type: premium, Total: 13
```

6406533042726. ✖

```
Type: premium, Total: 35  
Type: regular, Total: 15
```

6406533042727. ✖

Question Number : 20 Question Id : 640653903563 Question Type : MSQ Calculator : Yes

Correct Marks : 2 Max. Selectable Options : 0

Question Label : Multiple Select Question

Select all snippets of code which will create

list of `price` from `data` and store it in

`product_prices` . The value of

`product_prices` should be `[20, 15]`

Options :

```
product_prices = [price for (type, quantity, price) in data]
```

6406533042728. ✖

```
product_prices = []  
for elem in data:  
    product_prices.append(elem["price"])
```

6406533042729. ✔

6406533042730. ✓

```
product_prices = list(elem["price"] for elem in data)
```

6406533042731. ✗

```
product_prices = [elem[-1] for elem in data]
```

Sem1 English1

Section Id :	64065364115
Section Number :	2
Section type :	Online
Mandatory or Optional :	Mandatory
Number of Questions :	26
Number of Questions to be attempted :	26
Section Marks :	100
Display Number Panel :	Yes
Section Negative Marks :	0
Group All Questions :	No
Enable Mark as Answered Mark for Review and Clear Response :	No
Maximum Instruction Time :	0
Sub-Section Number :	1
Sub-Section Id :	640653133986
Question Shuffling Allowed :	No

Question Number : 21 Question Id : 640653903564 Question Type : MCQ Calculator : Yes
Correct Marks : 0

Question Label : Multiple Choice Question

THIS IS QUESTION PAPER FOR THE SUBJECT "FOUNDATION LEVEL : SEMESTER I: ENGLISH I (COMPUTER BASED EXAM)"

ARE YOU SURE YOU HAVE TO WRITE EXAM FOR THIS SUBJECT?

CROSS CHECK YOUR HALL TICKET TO CONFIRM THE SUBJECTS TO BE WRITTEN.

(IF IT IS NOT THE CORRECT SUBJECT, PLS CHECK THE SECTION AT THE TOP FOR THE SUBJECTS REGISTERED BY YOU)

Options :

6406533042732. ✓ YES

6406533042733. ✗ NO

Sub-Section Number : 2