

Consider a simple neural network with one hidden layer. The network has the following architecture:

- Input layer with 3 neurons.
- Hidden layer with 2 neurons, using the sigmoid activation function.
- Output layer with 1 neuron, using the linear activation function.

The weights and biases for the network are as follows:

Hidden Layer:

Neuron 1:
Weights: [0.5, -0.2, 0.8]
Bias: 0.1

Neuron 2:
Weights: [-0.3, 0.6, -0.7]
Bias: -0.4

Output Layer:

Neuron 1:
Weights: [0.2, 0.4]
Bias: -0.3

Assume that the input values are [0.6, 0.3, 0.8].

Calculate output of Neuron 1 in hidden layer

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Range

Text Areas : PlainText

Possible Answers :

0.60 to 0.80

AppDev1	
Section Id :	64065341309
Section Number :	6
Section type :	Online

Mandatory or Optional :	Mandatory
Number of Questions :	33
Number of Questions to be attempted :	33
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 :	Yes
Maximum Instruction Time :	0
Sub-Section Number :	1
Sub-Section Id :	64065388044
Question Shuffling Allowed :	No
Is Section Default? :	null

Question Number : 117 Question Id : 640653611034 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 0

Question Label : Multiple Choice Question

THIS IS QUESTION PAPER FOR THE SUBJECT "DIPLOMA LEVEL : MODERN APPLICATION DEVELOPMENT 1 (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 :

6406532040652.  YES

6406532040653.  NO

Sub-Section Number : 2
Sub-Section Id : 64065388045
Question Shuffling Allowed : Yes
Is Section Default? : null

Question Number : 118 Question Id : 640653611035 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

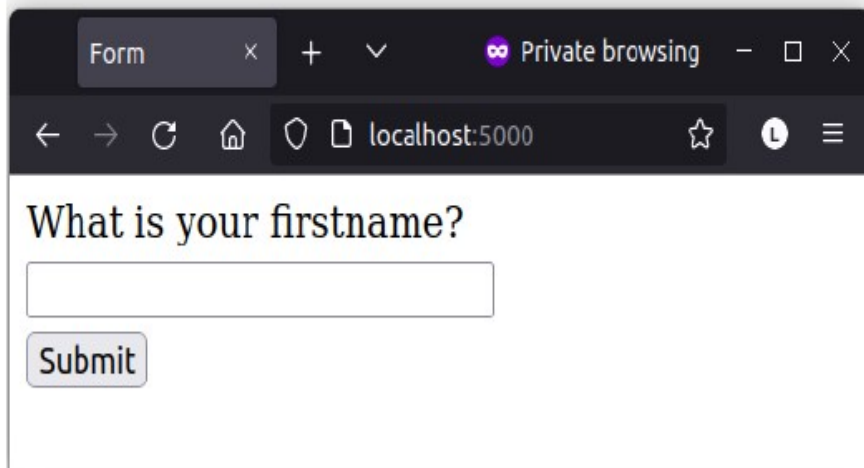
Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following snippet of HTML and its output rendered on the browser:

index.html

```
<form action="/">
  <label for="fname">What is your first name?</label><br>
  <input type="text" name="fname" id="fname"><br>
  <input type="submit" value="Submit">
</form>
```



Form x + ▾ Private browsing - □ ×

← → ↻ 🏠 🛡️ 📄 localhost:5000 ☆ ⓘ ☰

What is your firstname?

Which HTTP request would be sent to the server if the user enters "Mahesh" into the text box and clicks submit?

Options :

6406532040654. ✓ GET

6406532040655. ✗ POST

6406532040656.

✖ PUT

6406532040657. ✖ DELETE

Question Number : 119 Question Id : 640653611037 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

A 6000 rpm magnetic disk is being used to store data. The disk can only spin in one direction and is operating at 6000 revolutions per minute. If the operating system sends a request to the disk controller to fetch data from the disk, but by the time the request reached the disk controller, the information had already moved 35 degrees passed the reader, what will be the latency (in milliseconds) before it can start retrieving data?

Options :

6406532040662. ✖ 0.97

6406532040663. ✔ 9.03

6406532040664. ✖ 90.3

6406532040665. ✖ 97

Question Number : 120 Question Id : 640653611039 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following flask resource created using flask_restful.

```
from flask import Flask, request
from flask_restful import Api, Resource, reqparse

app = Flask(__name__)
api = Api(app)

parser = reqparse.RequestParser()
parser.add_argument("val")

class RestApi(Resource):
    def post(self, val):
        arg1 = parser.parse_args()
        arg2 = request.args
        return {
            "value_1": arg1["val"],
            "value_2": arg2["val"],
            "value_3": val
        }

api.add_resource(RestApi, "/api/games/<val>")
app.run(debug = True)
```

If the application is running locally on `http://127.0.0.1:5000`, What will be the output on the terminal for the command:

```
curl http://127.0.0.1:5000/api/games/cricket?val=soccer -d
{"val\":\"tennis\"}" -X POST -H "Content-Type: application/json"
```

Options :

```
{
    "value_1": "soccer",
    "value_2": "tennis",
    "value_3": "cricket"
}
```

6406532040670. ✖

6406532040671. ✖

```
{  
    "value_1": "cricket",  
    "value_2": "soccer",  
    "value_3": "tennis"  
}
```

```
{  
    "value_1": "tennis",  
    "value_2": "soccer",  
    "value_3": "cricket"  
}
```

6406532040672. ✓

```
{  
    "value_1": "tennis",  
    "value_2": "cricket",  
    "value_3": "soccer"  
}
```

6406532040673. ✖

Question Number : 121 Question Id : 640653611042 Question Type : MCQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following function to be tested and test functions given in the Python code snippet below.

test_file.py

```
import pytest

def square(x):
    sum = 0
    for counter in range(x):
        sum += x
    return sum

@pytest.mark.marker1
def testcase_1():
    assert square(11) == 121

@pytest.mark.marker2
def testcase_2():
    assert square(8) == 8

@pytest.mark.marker3
def testcase_3():
    assert square(5) == 36
```

On running this file on the terminal using pytest, the summary of the output is;

```
===== 1 passed, 2 deselected, 3 warnings in 0.02s =====
```

What command will result into the outcome given above?

Options :

6406532040682. ✖ `pytest test_file.py -k marker1`

6406532040683. ✖ `pytest test_file.py -m marker1`

6406532040684. ✔ `pytest test_file.py -m marker2`

6406532040685. ✖

```
pytest test_file.py -m marker3
```

Question Number : 122 Question Id : 640653611043 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

The hexadecimal equivalent of the IPv4 address 192.168.72.101 would be

_____.

Options :

6406532040686. ✖ C048 A865

6406532040687. ✔ C0A8 4865

6406532040688. ✖ C0A8 6548

6406532040689. ✖ C065 A848

Question Number : 123 Question Id : 640653611048 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

An HTML code and CSS code is given below. Which of the following correctly represents its rendered output?

CSS Code:

```
.one{color: blue;}  
.two{color: red !important;}  
.three{color: green;}
```


HTML Code:

```
<!DOCTYPE html>
<html>
<head>
  <title>Document</title>
  <link href="style.css" rel="stylesheet">
  <style>
    body{background-color: aqua; text-align: center;}
    h2{color: violet !important ;}
  </style>
</head>
<body style="background-color:lavender;" >
  <h2 style="color: blue;">Welcome to IIT</h2>
  <p class="one">My content 1 </p>
  <p class="two" style="color: brown;" >My content 2</p>
  <p class="three">My content 3</p>
</body>
</html>
```

Options :



6406532040706. ✖



6406532040707. ✖



6406532040708. ✖

Welcome to IIT

My content 1

My content 2

My content 3

6406532040709. ✓

Question Number : 124 Question Id : 640653611049 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following models Student and Project corresponding to tables student and project in SQLite database.

```
class Student(db.Model):
    id = db.Column(db.Integer(), primary_key = True)
    name = db.Column(db.String(), unique = True)

class Project(db.Model):
    id = db.Column(db.Integer(), primary_key = True)
    title = db.Column(db.String(), unique = True)
    course = db.Column(db.Integer(), db.ForeignKey("course.id"))
```

Based on the model schemas, what relationship do the classes Student and Project share?

Options :

6406532040710. ✖ Many-to-Many

6406532040711. ✖ One-to-Many

6406532040712. ✖ One-to-One

6406532040713. ✓ The tables are not at all related

Question Number : 125 Question Id : 640653611050 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following Python code snippet.

```
from flask import Flask
from flask_restful import Api, Resource

app = Flask(__name__)
api = Api(app)

class MyApi(Resource):
    def get(self, get):
        return {"greet": "Hello from GET Api!"}

    def put(self):
        return {"greet": "Hello from PUT Api!"}

api.add_resource(MyApi, '/api/<get>', '/api/put')

app.run()
```

If this application is running locally on `http://127.0.0.1:5000`, which of the following curl commands will throw an error?

1. `curl http://127.0.0.1:5000/api/get -X get`
2. `curl http://127.0.0.1:5000/api/put -X put`
3. `curl http://127.0.0.1:5000/api/post -X get`
4. `curl http://127.0.0.1:5000/api/get -X put`

Options :

6406532040714. ✖ Only 2 and 4

6406532040715. ✔ Only 4

6406532040716. ✖ Only 3 and 4

6406532040717. ✖ Only 3

Question Number : 126 Question Id : 640653611052 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the code below and match the conditions in Column A with respect to the coverage types in Column B.

```
int foo(int x, int y)
{
    int z = 0;
    if((x > 0) && (y > 0))
    {
        z = x;
    }
    return z;
}
```

Column A	Column B
1. Test invokes foo() at least once	a. Condition coverage
2. foo(1,1)	b. Branch coverage
3. foo(1,1) and foo(1,0)	c. Function coverage
4. foo(0,1) and foo(1,0)	d. Statement coverage

Options :

6406532040722. ✖ 1-d, 2-a, 3-b, 4-c

6406532040723. ✔ 1-c, 2-d, 3-b, 4-a

6406532040724. ✖ 1-c, 2-d, 3-a, 4-b

6406532040725. ✖ 1-b, 2-d, 3-c, 4-a

Question Number : 127 Question Id : 640653611053 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Match the following types of testing with their functionality.

A. Regression testing	1. Beta Testing
B. User Acceptance testing	2. One step beyond integration Testing
C. System testing	3. Simulates actual user interaction, allows to script browser
D. System testing Automation	4. Type of testing that runs after every change to ensure that the change introduces no unintended breaks.

Which of the following is the correct matching?

Options :

6406532040726. ✖ A → 1, B → 2, C → 3, D → 4

6406532040727. ✖ A → 4, B → 3, C → 2, D → 1

6406532040728. ✖ A → 3, B → 2, C → 1, D → 4

6406532040729. ✔ A → 4, B → 1, C → 2, D → 3

Question Number : 128 Question Id : 640653611056 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following table "emp" created in SQLite database corresponding to model class "Employee" using flask_sqlalchemy.

Id	Name	Designation	Gender	Salary
1	Raji	Headmaster	Female	4500
2	Ram	Teacher	Male	5000
3	Raveena	Teacher	Female	3000
4	Reshma	Technical staff	Female	2000
5	Ravi	Teacher	Male	1000

Which of the following code snippets will change the designation of male teachers to professor and will increase their salaries to 6000 correctly when typed in the Python console?

Options :

```
>>> emp = Employee.query.filter_by(Designation = 'Teacher').all()
>>> for staff in emp:
...     staff.Designation="Professor"
...     staff.Salary = 6000
...
>>> db.session.commit()
```

6406532040735. ✖

```
>>> emp = Employee.query.filter_by(Designation="Teacher", Gender="Male").all()
>>> for staff in emp:
...     staff.Designation="Professor"
...     staff.Salary +=6000
...
>>> db.session.commit()
```

6406532040736. ✖

```
>>> emp = Employee.query.filter_by(Designation = 'Teacher').all()
>>> for staff in emp:
...     staff.Designation="Professor"
...     staff.Salary += 6000
...
>>> db.session.commit()
```

6406532040737. ✖

6406532040738. ✔

```
>>>
emp=Employee.query.filter_by(Designation="Teacher",Gender="Male").all()
>>> for staff in emp:
...     staff.Designation="Professor"
...     staff.Salary=6000
...
>>> db.session.commit()
```

Question Number : 129 Question Id : 640653611058 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following python code snippet app.py, the html files, base.html and home.html residing in “templates” folder.

app.py

```
from flask import Flask, render_template
app = Flask(__name__)
@app.route('/')
def home():
    return render_template('home.html')
app.run(debug=True)
```

home.html

```
{% extends "base.html" %}
{% block content %}
<h3>Welcome to MAD I</h3>
{% endblock %}
```

base.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>IITM</title>
</head>
<body>
    <h1 style="color: violet;"> IITM BS Degree </h1>
    {% block content %}
    {% endblock %}
</body>
</html>
```

What will be the rendered output for base URL if flask app is running locally on <http://localhost:5000>

Options :

6406532040743. ✖ **Welcome to MAD I**

Welcome to MAD I

6406532040744. ✖ **IITM BS Degree**

IITM BS Degree

6406532040745. ✔ **Welcome to MAD I**

6406532040746. ✖ Error

Question Number : 130 Question Id : 640653611065 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following SQL create statement.

```
CREATE TABLE car (  
    car_id INTEGER NOT NULL,  
    model INTEGER NOT NULL,  
    name VARCHAR(50),  
    mfd_date DATETIME NOT NULL,  
    description VARCHAR,  
    PRIMARY KEY (car_id),  
    UNIQUE (model),  
    UNIQUE (name)  
)
```

Which of the following flask_sqlalchemy models will create exactly the same table as created by the above SQL command?

Options :

```
class Car(db.Model):  
    car_id = db.Column(db.Integer(), unique = True, nullable = False)  
    model = db.Column(db.Integer(), autoincrement = True)  
    name = db.Column(db.String(50), unique = True, nullable = True)  
    mfd_date = db.Column(db.DateTime(), nullable = False)  
    description= db.Column(db.String())
```

6406532040771. ✖

6406532040772. ✖

```
class Car(db.Model):
    car_id = db.Column(db.Integer(), primary_key = True)
    model = db.Column(db.Integer(), unique = True, nullable = True)
    name = db.Column(db.String(50), unique = False)
    mfd_date = db.Column(db.DateTime(), nullable = False)
    description = db.Column(db.String())
```

```
class Car(db.Model):
    car_id = db.Column(db.Integer(), primary_key = True)
    model = db.Column(db.Integer(), unique = True, nullable = False)
    name = db.Column(db.String(), unique = True, nullable = True)
    mfd_date = db.Column(db.DateTime())
    description = db.Column(db.String(50))
```

6406532040773. ✖

```
class Car(db.Model):
    car_id = db.Column(db.Integer(), primary_key = True)
    model = db.Column(db.Integer(), unique = True, nullable = False)
    name = db.Column(db.String(50), unique = True, nullable = True)
    mfd_date = db.Column(db.DateTime(), nullable = False)
    description = db.Column(db.String())
```

6406532040774. ✔

Sub-Section Number :	3
Sub-Section Id :	64065388046
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 131 Question Id : 640653611036 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Multiple Choice Question

Consider the following statements and select the correct option.

Statement 1: It is not necessary to implement server-side validation if client-side validation has been implemented.

Statement 2: Client-side validation can be implemented with HTML5.

Options :

6406532040658. ✖ Statement 1 is true while statement 2 is false

6406532040659. ✔ Statement 2 is true while statement 1 is false

6406532040660. ✖ Both statements are true

6406532040661. ✖ Both statements are false

Question Number : 132 Question Id : 640653611041 Question Type : MCQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Multiple Choice Question

Consider the following statements and select the correct answer.

Statement 1: In PaaS, developers are provided with an up-to-date development environment with the languages and frameworks of their choice.

Statement 2: Microsoft Azure provides IaaS as well as PaaS services.

Options :

6406532040678. ✖ Statement 1 is true, but statement 2 is false

6406532040679. ✖ Statement 2 is true, but statement 1 is false

6406532040680. ✖ Both statements are false

6406532040681. ✔ Both statements are true

Question Number : 133 Question Id : 640653611047 Question Type : MCQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Multiple Choice Question

Consider the following statements and choose the correct option.

Statement 1: If we submit the form using the GET method, then the form data is appended inside the body of the HTTP request.

Statement 2: Never use GET method in form to send sensitive data while it is good for non-secure data, like query strings in Google

Options :

6406532040702. ✖ Both statements 1 and 2 are true, and statement 2 is a correct explanation of statement 1

6406532040703. ✖ Both statements 1 and 2 are true, but statement 2 is not a correct explanation of statement 1

6406532040704. ✖ Statement 1 is true and statement 2 is false

6406532040705. ✔ Statement 2 is true and statement 1 is false

Question Number : 134 Question Id : 640653611051 Question Type : MCQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Multiple Choice Question

Which of the following git command will remove a file from the staged area but keeps the file in the directory?

Options :

6406532040718. ✖ `git rm <filename>`

6406532040719. ✖ `git del <filename>`

6406532040720. ✖ `git del --cached <filename>`

6406532040721. ✔ `git rm --cached <filename>`

Sub-Section Number : 4
Sub-Section Id : 64065388047
Question Shuffling Allowed : Yes
Is Section Default? : null

Question Number : 135 Question Id : 640653611038 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3 Max. Selectable Options : 0

Question Label : Multiple Select Question

Consider the following python code snippet.

```
from string import Template

statement = "The four major operations on the database are $c, $r, $u
and $d."

temp = Template(statement)

print(=== OUTPUT ===)
```

Which of the following statements, when substituted in place of `=== OUTPUT ===`, will not throw a `KeyError`?

Options :

6406532040666. ✓

```
temp.safe_substitute({"c": "CREATE", "r": "READ", "u": "UPDATE",
"d": "DELETE"})
```

6406532040667. ✓

```
temp.safe_substitute({"c": "CREATE", "r": "READ", "u": "UPDATE"})
```

6406532040668. ✓

```
temp.safe_substitute({"c": "CREATE", "r": "READ", "u": "UPDATE",
"d": "DELETE", "e": "TRUNCATE"})
```



```
temp.substitute({"c": "CREATE", "r": "READ", "u": "UPDATE",  
"d": "DELETE", "e": "TRUNCATE"})
```

6406532040669. ✓

Question Number : 136 Question Id : 640653611063 Question Type : MSQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3 Max. Selectable Options : 0

Question Label : Multiple Select Question

Consider the following `TodoSimple` resource class created using `flask_restful`.

```
from flask import Flask, request  
from flask_restful import Resource, Api  
  
app = Flask(__name__)  
api = Api(app)  
  
class TodoSimple(Resource):  
    def get(self, todo_id):  
        return {"todo_id": todo_id}  
  
    def put(self):  
        todo_id = request.args.get("todo_id")  
        return {"todo_id": todo_id}  
  
app.run()
```

Which of the following statements given below will correctly map the resource URLs with the `TodoSimple` resource class.

Options :

```
api.add_resource(TodoSimple, "/api")
```

6406532040763. ✖

```
api.add_resource(TodoSimple, "/api/<int:todo_id>", "/api")
```

6406532040764. ✓

6406532040765. ✖

```
api.add_resource(TodoSimple, "/api/<int:todo_id>")
```

6406532040766. ✔

```
api.add_resource(TodoSimple, "</int:todo_id>", "/api")
```

Sub-Section Number : 5
Sub-Section Id : 64065388048
Question Shuffling Allowed : Yes
Is Section Default? : null

Question Number : 137 Question Id : 640653611040 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2 Max. Selectable Options : 0

Question Label : Multiple Select Question

Which of the statements are true?

Options :

6406532040674. ✖ HTML5 is based on SGML

6406532040675. ✔ XHTML is based on XML which in turn is based on SGML

6406532040676. ✖ HTML5 is not backwards compatible with older versions of HTML

6406532040677. ✔ XML is both human and machine-readable

Question Number : 138 Question Id : 640653611046 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2 Max. Selectable Options : 0

Question Label : Multiple Select Question

Choose the correct code snippet to implement logging by importing python logging module

Options :

6406532040698. ✓

```
import logging
logging.basicConfig(filename='debug.log', level=logging.DEBUG,
format=f'%(asctime)s %(levelname)s %(name)s %(threadName)s :
%(message)s')
```

6406532040699. ✓

```
import logging
logging.basicConfig(filename='debug.log', level=logging.NOTSET,
format=f'%(asctime)s %(levelname)s %(name)s %(threadName)s :
%(message)s')
```

6406532040700. ✓

```
import logging
logging(filename='debug.log', level=logging.DEBUG)
```

6406532040701. ✓

```
import logging
logging.basicConfig(filename='debug.log')
```

Question Number : 139 Question Id : 640653611059 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2 Max. Selectable Options : 0

Question Label : Multiple Select Question

Consider a typical Amazon Alexa device. Which of the following would constitute the view of the application behind such a device?

Options :

6406532040747. ✓ The AI voice

6406532040748. ✗ The body of the device

6406532040749. ✓ The LED light around the device

6406532040750. ✗ None of these

Question Number : 140 Question Id : 640653611060 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2 Max. Selectable Options : 0

Question Label : Multiple Select Question

Which of the following is true of “cold storage” like Amazon Glacier?

Options :

6406532040751. ✔ They have low cost and high durability.

6406532040752. ✖ They have high cost and low durability.

6406532040753. ✖ Latency of retrieval is very low.

6406532040754. ✔ Latency of retrieval is very high.

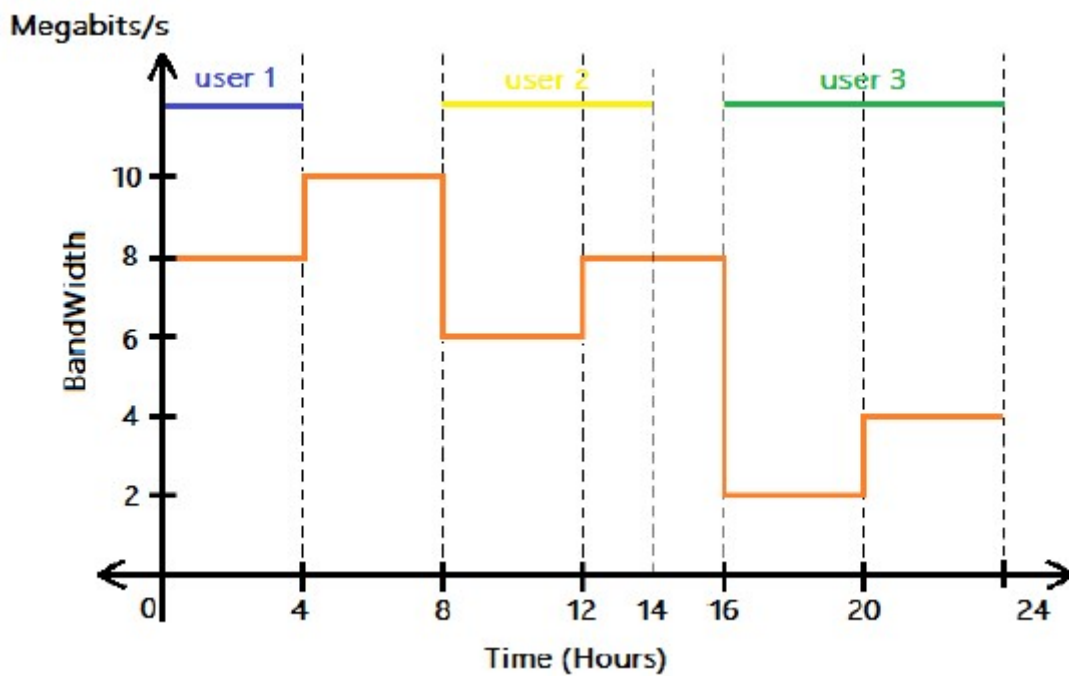
Sub-Section Number :	6
Sub-Section Id :	64065388049
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 141 Question Id : 640653611044 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4.5

Question Label : Multiple Choice Question

Consider the following graph that represents the variation in bandwidth of a network for an entire day (24 hours). Three users were connected to the network at three different times of the day. Which user has consumed the highest amount of data?



Options :

6406532040690. ✖ User 1

6406532040691. ✔ User 2

6406532040692. ✖ User 3

6406532040693. ✖ All the three users consumed same amount of data

Question Number : 142 Question Id : 640653611045 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4.5

Question Label : Multiple Choice Question

Consider the following restful implementation using flask.

```

from flask import Flask, request
from flask_restful import Resource, Api

app = Flask(__name__)
api = Api(app)

student_info = {"name": "Rahul", "Roll No": "12d2000678",
"Email": "onlinedegree.iitm.ac.in"}

class Student(Resource):
    def put(self):
        data = request.json
        student_info.update(data)
        return student_info

    def delete(self):
        student_info.popitem()
        return "success", 200

api.add_resource(Student, '/')

app.run()

```

If the above application is running on <http://127.0.0.1:5000> then what will be the final output on terminal on executing these two curl commands in the order mentioned?

1: `curl -X DELETE 'http://127.0.0.1:5000'`

2: `curl -X PUT -H "Content-Type: application/json" -H "Accept-Type: application/json" -d '{"Roll No": "12f2000555", "Email": "ds.study.iitm.ac.in"}' 'http://127.0.0.1:5000/'`

Options :

6406532040694. ✓ {"name": "Rahul", "Roll No": "12f2000555", "Email": "ds.study.iitm.ac.in"}

6406532040695. ✗ {"name": "Rahul", "Roll No": "12f2000555"}

6406532040696. ✗ {"Roll No": "12f2000555", "Email": "ds.study.iitm.ac.in"}

6406532040697. ✗ None of these

Question Number : 143 Question Id : 640653611055 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4.5

Question Label : Multiple Choice Question

Consider the following python code snippet. What will be the rendered output?

```
from jinja2 import Template
persons=[
    {"Gender":"Male", "Age":40, "Name":"John"},
    {"Gender":"Female", "Age":16, "Name":"Samantha"},
    {"Gender": "Male", "Age":20, "Name":"Kim"}
]

t="""
<ul>
{% for group in persons|groupby('Gender') %}
  <li>
    {{ group.grouper }}
    <ul>
      {% for person in group.list %}
        <li>{{ person.Name }} is {{ person.Age }} years old</li>
      {% endfor %}
    </ul>
  </li>
{% endfor %}
</ul>

"""
temp=Template(t)
print(temp.render(persons=persons))
```

Options :

- Male
 - John is 40 years old
- Female
 - Samantha is 16 years old
- Male
 - Kim is 20 years old

6406532040731. ✖

- Male
 - John is 40 years old
 - Kim is 20 years old
- Female
 - Samantha is 16 years old

6406532040732. ✖

- Female
 - Samantha is 16 years old
- Male
 - John is 40 years old
 - Kim is 20 years old

6406532040733. ✔

- Female
 - Samantha is 16 years old
- Male
 - John is 40 years old
- Male
 - Kim is 20 years old

6406532040734. ✖

Question Number : 144 Question Id : 640653611057 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4.5

Question Label : Multiple Choice Question

Consider a function `func`, and a set of test cases given below.

Filename: `test_file.py`

```
import pytest
def func(x,y):
    out = x+y**2
    return out

class Test_class0():
    def test_case1(self):
        assert func(1,2) == 5

    def case_test2(self):
        assert func(2,3) == 10

    def test_case3(self):
        assert func(6,2) == 8

class Test_class1():
    def test_case1(self):
        assert func(5,2) == 9

    def case_test2(self):
        assert func(4,3) == 14
```

What will be the output on the terminal for the command below?

`pytest test_file.py -k Test_class0`

Options :

6406532040739. ✖

`== 2 failed, 1 passed in 0.17s ==`

6406532040740. ✖

`== 2 failed, 1 deselected in 0.17s ==`

6406532040741. ✔

`== 1 failed, 1 passed, 1 deselected in 0.17s ==`

6406532040742. ✖

`== 1 failed, 2 passed in 0.17s ==`

Question Number : 145 Question Id : 640653611061 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4.5

Question Label : Multiple Choice Question

What will be the output of the following Python code snippet?

```
def decor(func):  
    def wrapper(x):  
        y=func(x)  
        return x*y  
    return wrapper  
  
@decor  
def output(x, optional="hello world!"):   
    return x, optional  
  
print(output(5))
```

Options :

6406532040755. ✖ ('hello world!', 'hello world!', 'hello world!', 'hello world!', 'hello world!')

6406532040756. ✖ (5, 5, 5, 5, 5)

6406532040757. ✔ (5, 'hello world!', 5, 'hello world!', 5, 'hello world!', 5, 'hello world!', 5, 'hello world!')

6406532040758. ✖ 25

Question Number : 146 Question Id : 640653611062 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4.5

Question Label : Multiple Choice Question

Consider the following flask application.

```
from flask import Flask
app = Flask(__name__)

@app.route('/user/<username>')
def show_user(username):
    return f'Hello {username} !'

@app.route("/hello/user")
def hello():
    return "Hello, user!"

@app.route("/user/")
def index():
    return "Hello user"

if __name__ == "__main__":
    app.run(debug=True)
```

If the application is running locally on <http://127.0.0.1:5000> then map the URLs in Column A with their rendered outputs in Column B.

	Column A		Column B
a	http://localhost:5000/hello/user	1	User hello !
b	http://localhost:5000/user/hello	2	Hello user !
c	http://localhost:5000/user/user	3	Hello hello !
d	http://localhost:5000/user/	4	Hello user
		5	Hello, user!
		6	Not Found

Options :

6406532040759. ✖ a-2, b-1, c-5, d-6

6406532040760. ✔ a-5, b-3, c-2, d-4

6406532040761. ✖ a-5, b-1, c-2, d-6

6406532040762. ✖ a-2, b-3, c-4, d-6

Question Number : 147 Question Id : 640653611064 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4.5

Question Label : Multiple Choice Question

Consider the following sorting algorithm which sorts any given unsorted array of numbers in ascending order. What would be its time complexity? (Assume appending and deleting an element from an array does not affect time complexity)

Step 1: Create an empty array.

Step 2: Find the element in the unsorted array with the minimum value.

Step 3: Append this element to the array created in step 1.

Step 4: Delete this element from the unsorted array.

Step 5: Repeat steps 1 to 4 until the unsorted array is empty.

Options :

6406532040767. ✔ $O(N^2)$

6406532040768. ✖ $O(N)$

6406532040769. ✖ $O(\log N)$

6406532040770. ✖ $O(N \log N)$

Sub-Section Number : 7

Sub-Section Id : 64065388050

Question Shuffling Allowed : Yes

Is Section Default? : null

Question Number : 148 Question Id : 640653611054 Question Type : SA Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Short Answer Question

Consider the following flask application.

```
from flask import Flask, render_template, request
app = Flask(__name__)

@app.route('/')
def out():
    val = request.args

    if val['num'] == '':
        return "<h1>Enter a valid number</h1>"
    elif val['num'].isalpha()==True:
        return "<h1>Invalid number</h1>"
    else:
        out = (val['num']) + (val['num'])
        return f'<h1>{out}</h1>'

if(__name__ == "__main__"):
    app.run(debug=True)
```

If this flask app is running locally on <http://localhost:5000>, what is the output for the following URL?

For input: <http://localhost:5000/?num=343>

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

343343

Sub-Section Number : 8

Sub-Section Id : 64065388051

Question Shuffling Allowed : Yes

Is Section Default? : null

Question Number : 149 Question Id : 640653611066 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4.5 Max. Selectable Options : 0

Question Label : Multiple Select Question

Follow the code given below

```
from flask import Flask, request
from flask_restful import Resource, Api, fields, marshal, marshal_with

app = Flask(__name__)
api = Api(app)

class User:
    def __init__(self, id, username, email):
        self.id=id
        self.username=username
        self.email=email

output={"id": fields.Integer,"username": fields.String,"email": fields.String}

#== Resource Class Here ==

api.add_resource(Userapi, '/')
if __name__ == '__main__':
    app.run()
```

What code should be written in place of #== Resource Class Here ==, so that we get;

```
{
    "id": 1,
    "username": "iitm",
    "email": "bs@ds.study.iitm.ac.in"
}
```

as output on running the command `curl -X GET 'http://127.0.0.1:5000'` in the terminal?

Options :

6406532040775. ✓


```
class Userapi(Resource):
    @marshal_with(output)
    def get(self):
        user=User(1,"iitm", "bs@ds.study.iitm.ac.in")
        return user
```

6406532040776. ✖

```
class Userapi(Resource):
    def get(self):
        user=User(1,"iitm", "bs@ds.study.iitm.ac.in")
        return user
```

```
@marshal_with(output)
class Userapi(Resource):
    def get(self):
        user=User(1,"iitm", "bs@ds.study.iitm.ac.in")
        return user
```

6406532040777. ✖

```
class Userapi(Resource):
    def get(self):
        user=User(1,"iitm", "bs@ds.study.iitm.ac.in")
        return marshal(user, output)
```

6406532040778. ✔

TDS

Section Id :	64065341310
Section Number :	7
Section type :	Online
Mandatory or Optional :	Mandatory
Number of Questions :	62
Number of Questions to be attempted :	62
Section Marks :	70
Display Number Panel :	Yes
Section Negative Marks :	0