6406531180609. ✖ No

Sub-Section Number : 2

Sub-Section Id : 64065351577

Question Shuffling Allowed : Yes

Question Number : 195 Question Id : 640653356111 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 correctly represents the components of the given URL?

```
https://www.mywebsite.com/home?user=Mad1&key=madcs2003
```

Options :

6406531180610. ✖
```
https : Domain name;
www.mywebsite.com : Request parameter;
/home : Directory;
user=Mad1&key=madcs2003 : domain name
```

6406531180611. ✖
```
https : Protocol;
www.mywebsite.com : Directory;
/home : Domain name;
user=Mad1&key=madcs2003 : Request parameters
```

6406531180612. ✔
```
https : Protocol;
www.mywebsite.com : Domain name;
/home : Directory;
user=Mad1&key=madcs2003 : Request parameters
```

6406531180613. ✖
```
https : IP Address;
www.mywebsite.com : Domain name;
/home : Directory;
user=Mad1&key=madcs2003 : Local Host
```

Question Number : 196 Question Id : 640653356113 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 flask app and an HTML file in templates folder:

Python file: app.py

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

my_list = ['Web development','onlinedegree','cs2003',
           'MAD-I','Data_science']

@app.route('/')
def render():
    return render_template('index.html', my_list = my_list)

app.run(debug = True)
```

Template file:

```
<!DOCTYPE html>
<head>
    <style>
        body{width: 200px;
             border: 2px solid black}
        #one{color:red;}
        #two{color:blue;}
    </style>
</head>
<body>
    {% for item in my_list %}
       {% set Length = item|length %}
       {% if Length%2 == 0 %}
            <h3 id = "one">{{ item }}</h3>
       {% else %}
            <h3 id = "two">{{ item }}</h3>
       {% endif %}
    {% endfor %}
</body>
```

If the above flask app is running locally on http://127.0.0.1:5000/, what will be rendered by the browser for the base URL?

Options :

6406531180618. ✖

6406531180619. ✖

Web development

onlinedegree

cs2003

MAD-I

Data_science

Web development

onlinedegree

cs2003

MAD-I

Data_science

6406531180620. ✖

Web development

onlinedegree

cs2003

MAD-I

Data_science

6406531180621. ✔

Question Number : 197 Question Id : 640653356129 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 flask code given below.

Python file: app.py

```
from flask import Flask, jsonify, request

app = Flask(__name__)

my_shops= [
  {
    'name of the shop' : 'Grocery',
    'items' : [
        {
            'item1' : 'Toothpaste',
            'item2' : 'Snacks',
            'item3' : 'Biscuits',
            'item4' : 'Soaps'
        }
    ]
  }
]

@app.route('/')
def show_shop():
    return jsonify({"shops" : my_shops})

#======================
     CODE HERE
#======================

if __name__ == '__main__':
    app.run()
```

Which of the following code snippets must be added in the given space of above application, in order to create a new shop in 'my_shops' list on the server side apart from the existing one?

Options :

6406531180671. ✖

```
@app.route('/myshop')
def create_shop():
    new_data = request.get_json()
    new_shop = [
        {
        'New shop' :  new_data['name of the shop']
        }
      ]
    my_shops[new_shop]
    return jsonify(new_shop)
```

6406531180672. ✔

```
@app.route('/myshop', methods=['POST'])
def create_shop():
    new_data = request.get_json()
    new_shop =  {
        'New shop' :  new_data['name of the shop']
       }
    my_shops.append(new_shop)
    return jsonify(new_shop)
```

6406531180673. ✖

```
@app.route('/myshop', methods=['POST'])
def create_shop():
    new_data = request.get_json()
    new_shop = {
         'New shop' :  new_data['name of the shop']
           }
    return jsonify(new_shop)
```

6406531180674. ✖ All of these

Question Number : 198 Question Id : 640653356132 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

A table 'person' is created in the database using model class "Person" with fields and their properties given in the table below.

id	firstname	lastname	email	age	occupation
1	Rahul	Mishra	rahul@gmail.com	23	Engineer
2	Ishan	Vadhera	vadhera@gmail.com	35	Lawyer
3	Abhilasha	Verma	vermaa@gmail.com	25	Teacher

Assuming that `flask_sqlachemy` is to be used in the 'main.py' file, which of the following statements is/are true?

Options :

6406531180683. ✖ Both the queries i.e., `Person.query.filter_by(firstname="Ishan").all()` and `Person.query.filter_by(firstname="Ishan").first()` will produce the same result.

6406531180684. ✔ Both the queries i.e., `Person.query.filter_by(id=3).first()` and `Person.query.get(3)` will produce the same result.

6406531180685. ✖ If `person1 = Person.query.get(1)` then, Both the inputs i.e., `>>>person1` `>>>person1.firstname` will produce the same result.

6406531180686. ✖ Both the queries i.e., `Person.query.filter_by(firstname="Ishan").first(), and Person.query.get(3)` will produce the same result

Question Number : 199 Question Id : 640653356133 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

A flask app and a template files are given below.

Python file: app.py

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

@app.route('/home')
def HomePage():
    return "Welcome, folks! This is the Home Page!"

@app.route('/about')
def AboutPage():
    users = [
            {"user":"Shobhit","gender" : "Male" , "age" : 23, "score": 90},
            {"user":"Deepak", "gender" : "Male" , "age" : 17, "score": 88},
            {"user":"Nikita", "gender" : "Female" , "age" : 20, "score": 87}
            ]
    return render_template('home.html', condition=True, users=users)

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

Templates file: home.html

```html
<!DOCTYPE html>
<html>
<body>
 <p><a href="{{ url_for('HomePage') }}">Go back to home page?</a></p>
<h2>About page</h2>
{% if condition %}
   <h3> You are landed on about page.</h3>
   {% for user in users %}
   <ul>
    <li>Username : {{user.user}}, Age : {{user.age}}, Gender :
    {{user.gender}}, Score : {{user.score}}</li>
   </ul>
   {% endfor %}
{% else %}
    <h3> Please Go back.</h3>
{% endif %}
</body>
</html>
```

If the above flask application is running locally on “http://127.0.0.1:5000”, which of the following statement is true?

Options :

6406531180687. ✖ For URL: “http://127.0.0.1:5000/home/”, the rendered output will be "Welcome, folks! This is the Home Page!"

6406531180688. ✔

For URL: “http://127.0.0.1:5000/about”, the rendered output will be:

Go back to home page?

About page

You are landed on about page.

- Username : Shobhit, Age : 23, Gender : Male, Score : 90
- Username : Deepak, Age : 17, Gender : Male, Score : 88
- Username : Nikita, Age : 20, Gender : Female, Score : 87

For URL: “http://127.0.0.1:5000/about”, the rendered output will be:

Go back to home page?

About page

You are landed on about page.

- Username : Shobhit
- Age : 23
- Gender : Male
- Score : 90

- Username : Deepak
- Age : 17
- Gender : Male
- Score : 88

- Username : Nikita
- Age : 20
- Gender : Female
- Score : 87

6406531180689. ✖

6406531180690. ✖

For URL: “http://127.0.0.1:5000/about”, the rendered ouptut will be:

Go back to home page?

About page

You are landed on about page.

Username : Shobhit

Age : 23

Gender : Male

Score : 90

Username : Deepak

Age : 17

Gender : Male

Score : 88

Username : Nikita

Age : 20

Gender : Female

Score : 87

Question Number : 200 Question Id : 640653356135 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

An HTML document is given below.

```
<!DOCTYPE html>
<html>
 <body>
   <h1 id="id1">Welcome to IITM</h1>
   <h3 class="class1">Welcome to the world's first online degree
        program.</h3>
   <a href="">Go back to main link</a>
   <p class="class1">Lorem ipsum dolor sit amet consectetur
       adipisicing elit. Earum, rerum?</p>
   <p class="class1">Have you enrolled in BSC in Data science
       and Programming? </p>
   <p id="id2">Go to the IITM online degree website and enroll
       now!</p>
  </body>
</html>
```

Suppose, if we want to give red color to the text within the heading element having id="id1" and green color to the text within the heading element having class="class1", what will be the correct way to do that?

Options :

6406531180695. ✖

By using external CSS as follows:

```
#id1{
    color:green;
}
h3.class1{
    color:red;
}
```

6406531180696. ✖

By using internal CSS as follows:

```
<style>
    .id1{
         color:red;
        }
    p.class1{
          color:green;
        }
</style>
```

6406531180697. ✔

By using inline CSS as follows:

```
<h1 id="id1" style="color:red;">Welcome to IITM</h1>
<h3 class="class1" style="color:green;">Welcome to the world's
first online degree program.</h3>
```

6406531180698. ✖ All of these

Question Number : 201 Question Id : 640653356140 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 HTML document with internal CSS.

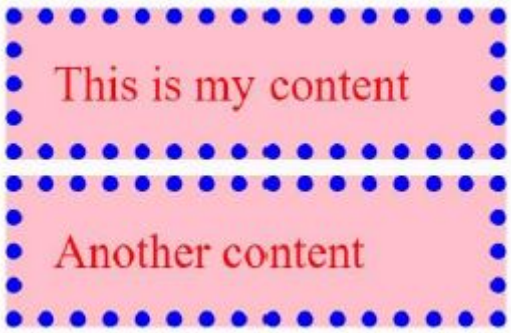
```
<!DOCTYPE html>
<html>
  <head>
      <style type="text/css">
         *{
           margin: 0px;
           width: 253px;
          }
         div{
              margin: 10px;
              padding: 20px;
              border-style: dotted;
              border-width: 10px;
              font-size: 30px;
              color: blue;
              background-color: pink;
              border-color: red;
             }
       </style>
       <title>End Sem</title>
   </head>
   <body>
      <div>This is my content</div>
      <div>Another content</div>
   </body>
</html>
```

How will the browser render the above HTML document?

Options :

6406531180715. ✖

6406531180716. ✖



6406531180717. ✖

6406531180718. ✔

Sub-Section Number : 3

Sub-Section Id : 64065351578

Question Shuffling Allowed : Yes

Question Number : 202 Question Id : 640653356114 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Multiple Select Question

Consider the following table "workers" created in SQLite database corresponding to model class "Workers" using `flask_sqlalchemy`.

Id	Name	Designation	Gender	Salary
Filter	Filter	Filter	Filter	Filter
1	Padma Raja	Supervisor	Female	2000
2	Sameer Gandhi	Labour	Male	1200
3	Latika Murthy	Labour	Female	800
4	Nitya Grover	Supervisor	Female	2000
5	Amit Saxena	Supervisor	Male	2000

The correct way to increase the salary of all the female workers by 500 Rupees using the Python console is:

Options :

6406531180622. ✖

```
>>> workers = Workers.query.filter_by(Designation =
'Supervisor').all()
>>> for worker in workers:
...     worker.Salary += 500
...
>>> db.session.commit()
```

6406531180623. ✖

```
>>> workers = Workers.query.filter_by(Gender = 'Female').all()
>>> workers.Salary += 500
>>> db.session.commit()
```

6406531180624. ✔

```
>>> workers = Workers.query.filter_by(Gender = 'Female').all()
>>> for worker in workers:
...     worker.Salary += 500
...
>>> db.session.commit()
```

6406531180625. ✔

```
>>> workers = Workers.query.filter(Workers.Gender.like('F%')).all()
>>> for worker in workers:
...     worker.Salary += 500
...
>>> db.session.commit()
```

Sub-Section Number : 4

Sub-Section Id : 64065351579

Question Shuffling Allowed : Yes

Question Number : 203 Question Id : 640653356112 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

How will the browser render the output of the following Python code snippet?

```
from jinja2 import Template

styles=[
        '.text{color: purple}\n #heading{color:red}\n #subhead{color:blue}',
        '.text{color: purple}\n #subhead{color:green}\n #main{color:blue}',
        '.text{color: purple}\n #main{color:red}\n #heading{color:blue}'
       ]

template = """
              <!DOCTYPE html>
              <style>
                 div{border: 2px solid black;
                     width: 300px;
                     background-color: rgb(247, 247, 230)}
                 {{styles[0]}}
              </style>
              <body>
                <div>
                  <h2 style="color:brown;" class="text" id="heading">
Programming Degree</h2>
                  <h3 class="text" id="subhead">Modern Application 1</h3>
                  <p class="text" id="main">This is a course on Application
Development</p>
                </div>
              </body>
           """
test_render = Template(template)
output = test_render.render(styles = styles)
print(output)
```

Options :

6406531180614. ✖

6406531180615. ✖

6406531180616. ✖

6406531180617. ✔

Question Number : 204 Question Id : 640653356118 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

What will be the output of the following python code if method `test_request_context()` allows flask app to print statements on the terminal?

```
from flask import Flask, url_for

app = Flask(__name__)

@app.route('/')
def home():
    return 'base url'

@app.route('/subscribe')
def subscribe():
    return 'Please subscribe to this page.'

@app.route('/new_course/<coursename>')
def course(coursename):
    return f'The course {coursename} gives basics of web development.'

with app.test_request_context():
    print(url_for('home'))
    print(url_for('subscribe'))
    print(url_for('subscribe', username = 'user_one'))
    print(url_for('course', coursename = 'MAD_I'))
```

Options :

6406531180634. ✖

```
base url
Please subscribe to this page
Please subscribe to this page user one.
The course MAD_I gives basics of web development.
```

6406531180635. ✖

```
/
/subscribe
/subscribe/user_one
/new_course/MAD_I
```

6406531180636. ✔

```
/
/subscribe
/subscribe?username=user_one
/new_course/MAD_I
```

6406531180637. ✖

```
/
/subscribe
/subscribe?username=user_one
/new_course?coursename=MAD_I
```

Question Number : 205 Question Id : 640653356126 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 internet connection with certain bandwidth is able to serve 10,000 requests of 150 Kilobytes each. What should be the increase in bandwidth (in Gbps) if this internet connection is to handle 12,500 requests of 180 Kilobytes each? (**Use relations:** 1 Byte = 8 bits, 1 MB = 1000 B, 1 GB = 1000 M and so on)

Options :

6406531180659. ✖ 600

6406531180660. ✔ 6

6406531180661. ✖ 0.6

6406531180662. ✖ 12

Question Number : 206 Question Id : 640653356127 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 snippet given below.

Python file: test_app.py

```
import pytest

@pytest.fixture
def items():
    return "Books"

@pytest.fixture
def order():
    return "Pens"

@pytest.fixture
def order_items(order, items):
    return [order, items]

@pytest.fixture
def expected_list():
    return ["Books", "Pencils", "Pens"]

def test_1(order_items, expected_list):
    order_items.append("Pencils")
    assert order_items == expected_list

def test_2(order_items):
    order_items.append("Pencils")
    assert order_items == ["Pens", "Books", "Pencils"]
```

Which of the following statement is true about the above code snippet?

Options :

6406531180663. ✔ After running pytest, test_1 will fail, whereas test_2 will pass.

6406531180664. ✖ After running pytest, test_2 will fail, whereas test_1 will pass.

6406531180665. ✖ Both the test cases, test_1 and test_2 will pass successfully.

6406531180666. ✖ Both the test cases, test_1 and test_2 will fail.

Question Number : 207 Question Id : 640653356130 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 Python code snippet given below.

Python file: app.py

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

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

class Add(Resource):
    def post(self):
        data_args = reqparse.RequestParser()
        data_args.add_argument('Name', help='Name is required',
        required =True)
        data_args.add_argument('Age', help='Age is required',
        required =True)
        args = data_args.parse_args()
        return { "Your Name": args['Name'], "Your Age" : args['Age']}

api.add_resource(Add, '/add')

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

If this flask application is running on http://127.0.0.1:5000, which of the following is the correct output when a POST request is sent to URL "http://127.0.0.1:5000/add"?

Options :

6406531180675. ✖ The server will throw a "405 METHOD NOT ALLOWED" error.

6406531180676. ✖ The server will throw a "404 NOT FOUND" error.

6406531180677. ✔ For the request body;

```
{
    "Name" : "Rahul",
    "Age": 23
}
```

The application will return;

```
{
    "Your Name": "Rahul",
    "Your Age": "23"
}
```

6406531180678. ✖ None of these

Question Number : 208 Question Id : 640653356131 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 Python code given below.

```
import pytest

@pytest.fixture
def first_entry():
    return "Apple"

@pytest.fixture
def order(first_entry):
    return [first_entry]

def test_string(order):
    order.append("Kiwi")
    assert order == ["Banana", "Apple"]

def test_int(order):
    order.append("Banana")
    assert order == ["Banana", "Apple", "Kiwi"]
```

Which of the following statement is true?

Options :

6406531180679. ✖ After running pytest, both test cases will pass successfuly.

6406531180680. ✖ After running pytest, the first test case will fail, whereas the second test case will pass.

6406531180681. ✔ After running pytest, both test cases will show a failure report.

6406531180682. ✖ None of these

Question Number : 209 Question Id : 640653356134 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 flask app given below.

```
from flask import Flask, abort
from flask_restful import Resource, Api
app = Flask(__name__)
api = Api(app)

item_list=[{"item1": "Cloths"},
           {"item2" : "Shoes"},
           {"item3" : "Sunglasses"}]

class ItemList(Resource):
    def get(self, item_no, item_name):
      this_item = {'item'+item_no : item_name}
      if this_item in item_list:
        return item_list, 200
      else:
        abort('400')

    def post(self, item_no, item_name):
      my_item = {'item'+item_no : item_name}
      item_list.append(my_item)
      return my_item, 201

api.add_resource(ItemList, '/items/<item_no>/<item_name>')
app.run(debug=True)
```

If the above flask application is running locally on "http://127.0.0.1:5000", what will be the output of a GET request sent to the URL: 'http://127.0.0.1:5000/items/4/watch' just after a POST request that is sent on the same URL?

Options :

6406531180691. ✖ The server will show a "404 NOT FOUND" error.

6406531180692. ✖

```
{
    "item4": "watch"
}
```

```
[
    {
        "item1": "Cloths"
    },
    {
        "item2": "Shoes"
    },
    {
        "item3": "Sunglasses"
    },
    {
        "item4": "watch"
    }
]
```

6406531180693. ✔

6406531180694. ✖ None of these

Question Number : 210 Question Id : 640653356139 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 application.

```
from flask import Flask, render_template

app=Flask(__name__)

@app.route('/')
def home():
    my_items = ['Cake', 'Apple', 'Ice Cream', 'DarkChocolate',
'Donut', 'Grape']
    l1 = []
    for i in range(len(my_items)):
        if i>2:
            l1.append(my_items[i])

    return render_template('index.html', list=l1)

app.run(debug=True)
```

Template File - index.html

```
{% macro display(list) %}
    {% for item in list %}
        <p>{{ item }}</p>
    {% endfor %}
{% endmacro %}
<html>
  <body>
     {{ display(list) }}
  </body>
</html>
```

suppose the application is running locally on the 'http://127.0.0.1:5000', then what will be rendered by the browser?

Options :

6406531180711. ✖ Cake

Apple

Ice Cream

6406531180712. ✖ Cake

Apple

Ice Cream

DarkChocolate

Donut

Grape

6406531180713. ✔ DarkChocolate

Donut

Grape

6406531180714. ✖ Ice Cream

DarkChocolate

Donut

Grape

Question Number : 211 Question Id : 640653356142 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 snippets.

File 1: main.py

```
import sys
from new import fun
a = sys.argv[1]
b = sys.argv[2]
c = sys.argv[3]
result = fun(a, b, c)
print(result + " is greater")
```

File 2: new.py

```
def fun(num1,num2,num3):
  if (num1 > num2) and (num1 > num3):
      return num1
  elif (num2 > num1) and (num2 > num3):
      return num2
  else:
      return num3
```

suppose the main.py file is executed in the terminal. What will be the output?

```
python main.py
python main.py 8 10 5
```

Options :

6406531180723. ✖ IndexError: list index out of range

NameError: name 'fun' is not defined

6406531180724. ✖ 3 is greater

NameError: name 'fun' is not defined

6406531180725. ✔ IndexError: list index out of range

10 is greater

6406531180726. ✖ NameError: name 'fun' is not defined

8 is greater

Question Number : 212 Question Id : 640653356144 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 a server that has an Intel i5 processor, 64 GB RAM, 1 TB Hard disk with 3 Gbps network connection. If a client accesses a web page, it requires 1.5 MB. Calculate the maximum number of

such requests per second the server can handle. (**Use relations:** 1 Byte = 8 bits, 1 MB = 1000 B, 1 GB = 1000 M and so on).

Options :

6406531180731. ✖ 25

6406531180732. ✖ 32

6406531180733. ✔ 250

6406531180734. ✖ 200

Sub-Section Number : 5

Sub-Section Id : 64065351580

Question Shuffling Allowed : Yes

Question Number : 213 Question Id : 640653356119 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Select Question

Consider the following Python code snippet.

```python
from flask import Flask, abort, redirect, url_for, render_template

app = Flask(__name__)

weekday_users = ['user_1','user_3','user_4','user_6','user_7']
weekend_users = ['user_2','user_5']

@app.route('/weekday/<username>')
def user_weekday(username):
    if username in weekday_users:
        return redirect(url_for('login', username = username))
    else:
        abort(401)

@app.route('/weekend/<username>')
def user_weekend(username):
    if username in weekend_users:
        return redirect(url_for('login', username = username))
    else:
        abort(401)

@app.route('/login/<username>')
def login(username):
    return f"<h2>Correct User Found! {username}</h2>"

@app.errorhandler(401)
def page_not_found(error):
    return "<h2>You are not authorized for this day.</h2>", 401

app.run()
```

If the above flask app is running locally on "http://127.0.0.1:5000", Which of the following statements is/are correct?

Options :

6406531180638. ✔ For the URL, "http://localhost:5000/weekday/user_4", the browser will render: **Correct User Found! user_4**

6406531180639. ✖ For the URL, "http://localhost:5000/weekend/user_3", the browser will render: **Correct User Found! user_3**

6406531180640. ✔ For the URL, "http://localhost:5000/weekday/user_5", the browser will render: **You are not authorized for this day.**

6406531180641. ✖ For the URL, “http://localhost:5000/weekend/user_2”, the browser will render:

You are not authorized for this day.

Question Number : 214 Question Id : 640653356136 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Select Question

Consider the code given below.

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

app = Flask(__name__)

api = Api(app)

Mytasks = {
    1: {"mytask": "Studying"},
    2: {"mytask": "Exercise"},
    3: {"mytask": "Eating"},
    4: {"mytask": "Sleeping"}
}
class Display(Resource):
    def get(self):
        return Mytasks

class DisplayAll(Resource):
    def get(self, MytaskList_id):
        return Mytasks[MytaskList_id]

    def post(self, MytaskList_id):
        data_args = reqparse.RequestParser()
        data_args.add_argument("mytask",help='This is required
field', required =True)
        args = data_args.parse_args()
        Mytasks[MytaskList_id] = {"mytask" : args["mytask"]}
        return Mytasks[MytaskList_id]
api.add_resource(Display, '/mytask')
api.add_resource(DisplayAll, '/task/<int:MytaskList_id>')

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

If the above flask application is running locally on "http://127.0.0.1:5000", which of the following statements is/are true?

Options :

6406531180699. ✔ The status code that we get after sending a POST request to the URL: 'http://127.0.0.1:5000/mytask/4' will be "404 NOT FOUND".

6406531180700. ✔

The response that we get after sending a GET request to the URL:http://127.0.0.1:5000/mytask" will be:

```
{
    "1": {
        "mytask": "Studying"
    },
    "2": {
        "mytask": "Exercise"
    },
    "3": {
        "mytask": "Eating"
    },
    "4": {
        "mytask": "Sleeping"
    }
}
```

The response that we get after sending POST request to the URL:"http://127.0.0.1:5000/task/4" with a sending a request body as,

```
{
    "mytask" : "Swimming"
}
will be :
{
    "mytask" : "Swimming"
}
```

6406531180701. ✔

6406531180702. ✖

The response that we get after sending a GET request to the URL:"http://127.0.0.1:5000/mytask" will be:


```
{
    "1": {
        "mytask": "Studying"
    },
    "2": {
        "mytask": "Exercise"
    },
    "3": {
        "mytask": "Eating"
    },
    "4": {
        "mytask": "Sleeping"
    },
    "4": {
        "mytask": "Swimming"
    }
}
```


Sub-Section Number : 6

Sub-Section Id : 64065351581

Question Shuffling Allowed : Yes

Question Number : 215 Question Id : 640653356123 Question Type : SA Calculator : None

Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4.5

Question Label : Short Answer Question

What will be the decimal representation of binary number 010101100000011_2?

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

11011

Sub-Section Number : 7

Sub-Section Id : 64065351582

Question Shuffling Allowed : Yes

Question Number : 216 Question Id : 640653356124 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

A machine takes a minimum of 100 seconds to sort 500 entries in a database. What will be the approximate minimum time taken by the machine to sort 1200 entries if the sorting method employs an algorithm with time complexity of O(nlog(n)). Where “n” is the number of entries?

Options :

6406531180651. ✖ 173 seconds

6406531180652. ✔ 273 seconds

6406531180653. ✖ 373 seconds

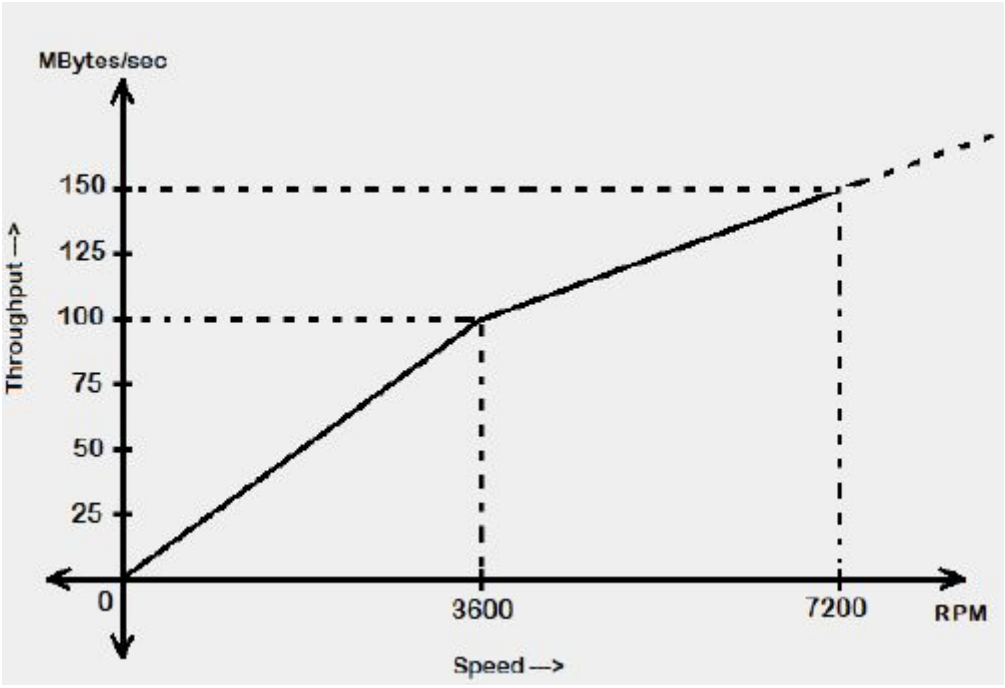
6406531180654. ✖ 473 seconds

Question Number : 217 Question Id : 640653356125 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

The speed vs. throughput characteristics of a typical HDD is shown in the figure below. If this HDD is to be used as a replacement of an SSD whose read/write speed is 450 MB/s. At what speed (in RPM) should the disk of HDD rotate with to deliver the same performance as that of the SSD?



Options :

6406531180655. ✖ 3600 RPM

6406531180656. ✖ 7200 RPM

6406531180657. ✖ 16,200 RPM

6406531180658. ✔ 28,800 RPM

Question Number : 218 Question Id : 640653356137 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 app and an HTML file.

Flask app: app.py

```
from flask import Flask, render_template, request
app = Flask(__name__)
users = {
          '3':{'name': 'Ram', 'Designation': 'Teacher'},
          '2':{'name': 'Dilip', 'Designation': 'student'},
          '5':{'name': 'Sonu', 'Designation': 'computer operator'},
          '1':{'name':'Guru', 'Designation': 'clerk'}
        }
@app.route('/')
def country():
    id = request.args.get('id')
    authenticated_users_id = [3, 2, 5]
    user = users.get(id)
    name = user.get('name') if user is not None else None
    Designation = user.get('Designation') if user is not None else
None
    user = {'is_authenticated': False, 'name': name, 'Designation':
Designation}
    if int(id) in authenticated_users_id:
        user['is_authenticated'] = True
        return render_template('index.html', data = user)
    if int(id) not in authenticated_users_id:
        user['is_authenticated'] = False
        return render_template('index.html', data = user)

app.run(debug = True)
```

HTML File: index.html

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <title>Document</title>
    </head>
    <body>
        {% if data.name == None %}
            User not found
        {% elif data.is_authenticated == True %}
            Hello {{data.name}} you can enter into this site:
{{data.Designation}}
        {% else %}
            Hello {{data.name}} you have  no access to this site:
{{data.Designation}}
        {% endif %}
     </body>
</html>
```

Suppose the application is running locally on the ‘http://127.0.0.1:5000', then what will be rendered by the browser for ‘http://127.0.0.1:5000/?id=5', http://127.0.0.1:5000/?id=1’ and ‘http://127.0.0.1:5000/?id=4' respectively?

Options :

6406531180703. ✔ Hello Sonu you can enter into this site: computer operator

Hello Guru you have no access to this site: clerk

User not found

6406531180704. ✖ Hello Ram you can enter into this site: Teacher

Hello Guru you have no access to this site: Student

Hello Sonu you can enter into this site: computer operator

6406531180705. ✖ User not found

Hello Dilip you have no access to this site: Student

Hello Sonu you can enter into this site: computer operator

6406531180706. ✖ Hello Ram you can enter into this site: Teacher

Hello Dilip you have no access to this site: Student

User not found

Question Number : 219 Question Id : 640653356138 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.

```
from jinja2 import Template

temp = """{% set numbers = studs | map(attribute = "mark") | list %}
          {{numbers | min}} {{numbers | max}}"""
studs = [
{"stud_name":"Reeta","mark":"92"},
{"stud_name":"Veena","mark":"88"},
{"stud_name":"Meena","mark":"62"},
{"stud_name":"uma","mark":"98"}
]
t1 = """{% for i in studs -%}
            {{i}}
        {%- endfor%}"""

output = Template(temp)
out = Template(t1)
print(output.render(studs = studs))
print(out.render(studs = studs))
```

What will be the output of the above program?

Options :

6406531180707. ✖ 98 62

{'stud_name': 'Reeta', 'mark': '92'}

{'stud_name': 'Veena', 'mark': '88'}

{'stud_name': 'Meena', 'mark': '62'}

{'stud_name': 'uma', 'mark': '98'}

6406531180708. ✔ 62 98

{'stud_name': 'Reeta', 'mark': '92'}{'stud_name': 'Veena', 'mark': '88'}{'stud_name': 'Meena', 'mark': '62'}{'stud_name': 'uma', 'mark': '98'}

6406531180709. ✖ 62 98

{'stud_name': 'uma', 'mark': '92'}

{'stud_name': 'Meena', 'mark': '88'}

{'stud_name': 'Veena', 'mark': '62'}

{'stud_name': 'Reeta', 'mark': '98'}

6406531180710. ✖ 98 62

{'stud_name': 'uma', 'mark': '92'}{'stud_name': 'Meena', 'mark': '88'}{'stud_name': 'Veena', 'mark': '62'}{'stud_name': 'Reeta', 'mark': '98'}

Question Number : 220 Question Id : 640653356141 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 table “newtable” in SQLite database.

ID	Name	Age	Mark	course
Filter	Filter	Filter	Filter	Filter
1	Vishnu	20	98	M1
2	Kumar	18	90	M2
3	Leela	20	90	M1
4	Naren	18	98	M2
5	Vishal	19	95	M1
6	Pranav	20	95	M2
7	Vinu	19	90	M1
8	Viki	18	95	M2

What will be the output of the following SQL queries given below?

```
CREATE UNIQUE INDEX IF NOT EXISTS index_name
ON newtable (Name ASC, Mark ASC) WHERE Age>18;
SELECT ID, Name, Age, Mark, course FROM newtable WHERE Age>18;
```

Options :

6406531180719. ✖

index_name will be created

ID	Name	Age	Mark	course
2	Kumar	18	90	M2
4	Naren	18	98	M2
8	Viki	18	95	M2

6406531180720. ✔

index_name will be created

ID	Name	Age	Mark	course
3	Leela	20	90	M1
6	Pranav	20	95	M2
7	Vinu	19	90	M1
5	Vishal	19	95	M1
1	Vishnu	20	98	M1

6406531180721. ✖

index_name will not be created

ID	Name	Age	Mark	course
2	Kumar	18	90	M2
4	Naren	18	98	M2
8	Viki	18	95	M2
3	Leela	20	90	M1
6	Pranav	20	95	M2
7	Vinu	19	90	M1
5	Vishal	19	95	M1
1	Vishnu	20	98	M1

6406531180722. ✖

index_name will not be created

ID	Name	Age	Mark	course
1	Vishnu	20	98	M1
2	Kumar	18	90	M2
3	Leela	20	90	M1
4	Naren	18	98	M2

Question Number : 221 Question Id : 640653356143 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 snippet.

```
@app.route("/search", methods = ["GET"])
def search():
   q= request.args.get('q')
   query = "%"+q+"%"
   results = Details.query.filter(Details.Content.like(query)).all()
   Return render template("result.html", q=q, Details = results)
```

File : result.html

```
{% for item in results %}
     {{item["Title"]}}
{% endfor %}
```

SQLite table: details

	ID	Title	Content
1	1	Introduction	Java is a powerful general-purpose ...
2	2	learn python	Python is a powerful general-purpose ...
3	3	Basics	Java works on different platforms (Windo
4	4	Code	Python is currently the most widely used

If the flask application is running locally on URL: http://127.0.0.1:5000, what will be rendered by the web browser for URL http://127.0.0.1:5000/search/q=Java?

Options :

6406531180727. ✖ Introduction

Code

6406531180728. ✖ Java is a powerful general purpose ...

Java works on different platforms(windows)

6406531180729. ✖ Introduction - Java is a powerful general purpose ...

learn python - Python is a powerful general purpose ...

Basics - Java works on different platforms(windows)

6406531180730. ✔ Introduction

Basics

Sub-Section Number : 8

Sub-Section Id : 64065351583

Question Shuffling Allowed : Yes

Question Number : 222 Question Id : 640653356128 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

Question Label : Multiple Select Question

Consider the HTML code given below.

HTML file: index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
</head>
<body>
<h1>
   <h1>CSS Selectors</h1>
   </h1>
    <div>
       <h2 id="header-id1" class="header-class">Hi, Folks!</h2>
       <p id="paragraph-id1" class="paragraph-class">Welcome to IITM.</p>
       <a href="https://iitm.ac.in">
        IIT Bsc Degree Website</a>
       <p class="class1 class2">We are launching World's first online
degree course.</p>
    </div>
    <div>
       <h2 id="header-id2">Have you enrolled to the program?</h2>
       <p id="paragraph-id2">If not, enroll now!</p>
       <a href="https://mywebsite.com
        ">Link to Enroll</a>
       <p class="class2 class3">Happy Learning!</p>
     </div>
</body>
</html>
```

CSS Selectors

Hi, Folks!

Welcome to IITM.

IIT Bsc Degree Website

We are launching World's first online degree course.

Have you enrolled to the program?

If not, enroll now!

Link to Enroll

Happy Learning!

To obtain the output as given in figure above, which of the following snippets of CSS code must be used?

Options :

6406531180667. ✖

```
<style>
    #header-id1, #header-id2, #header-id3
    {
        color: red;
    }
    .header-class{
        color:blue;
    }
    .paragraph-class{
        color:purple;
    }
    p.class2{
       color:green;
    }
</style>
```

6406531180668. ✔

```
<style>
    #header-id1, #header-id2, #header-id3
    {
    color: red;
    }
    .header-class{
        color:blue;
    }
    p.class2{
       color:green;
    }
    #paragraph-id1, #paragraph-id2{
       color:purple;
    }
</style>
```

6406531180669. ✖

```
<style>
    h2{
        color: red;
    }
    .header-class{
        color:blue;
    }
    p.class3{
       color:green;
    }
    #paragraph-id{
       color:purple;
    }
</style>
```

```
<style>
    h2{
    color: blue;
    }
    h2{
    color: red;
    }
    #paragraph-id1, #paragraph-id2{
       color:purple;
    }
    p.class2{
       color:green;
    }
</style>
```

6406531180670. ✔

Sub-Section Number : 9

Sub-Section Id : 64065351584

Question Shuffling Allowed : No

Question Id : 640653356115 Question Type : COMPREHENSION Sub Question Shuffling Allowed : No Group Comprehension Questions : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Question Numbers : (223 to 224)

Question Label : Comprehension

Consider the following model classes “State” and “City” corresponding to tables “state” and “city”

respectively in the SQLite database.

```
class State(db.Model):
      state_id = db.Column(db.Integer(), primary_key = True)
      state_name = db.Column(db.String(50), nullable = False)
      cities = db.relationship("City", backref = "stateof")

class City(db.Model):
      city_id = db.Column(db.Integer(), primary_key = True)
      city_name = db.Column(db.String(50), nullable = False)
      state = db.Column(db.Integer(), db.ForeignKey("state.state_id"))
```

Based on the above data, answer the given subquestions.

Sub questions

Question Number : 223 Question Id : 640653356116 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Select Question

If an object "s1" that represents an existing record in the table "state" is defined as `s1 = State.query.get(1)`,
The correct way(s) to add a city with the name "Chennai" that belongs to s1 using the Python console is.

Options :

6406531180626. ✖

```
>>> c1 = City(city_name = "Chennai", state = s1)
>>> db.session.add(c1)
>>> db.session.commit()
```

6406531180627. ✔

```
>>> c1 = City(city_name = "Chennai", state = 1)
>>> db.session.add(c1)
>>> db.session.commit()
```

6406531180628. ✔

```
>>> c1 = City(city_name = "Chennai", stateof = s1)
>>> db.session.add(c1)
>>> db.session.commit()
```

6406531180629. ✖

```
>>> c1 = City(city_name = "Chennai", stateof = 1)
>>> db.session.add(c1)
>>> db.session.commit()
```

Question Number : 224 Question Id : 640653356117 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Select Question

If “s1” and “c1” are existing objects in the tables “state” and “city” respectively where:

```
s1 = State.query.filter_by(state_name = "Maharashtra").first()
```

and

```
c1 = City.query.filter_by(city_name = "Mumbai").first()
```

Which of the following statements is/are correct?

Options :

6406531180630. ✖ The input `s1.cities` on Python console will return a single object

6406531180631. ✔ The input `s1.cities` on Python console will return a list of object(s)

6406531180632. ✔ The input `c1.state` on Python console will return a single object

6406531180633. ✖ The input `c1.state` on Python console will return a list of object(s)

Question Id : 640653356120 Question Type : COMPREHENSION Sub Question Shuffling

Allowed : No Group Comprehension Questions : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Question Numbers : (225 to 226)

Question Label : Comprehension

Consider the following resource API for the employee information given below and answer the given subquestions.

```
from flask import Flask
from flask_restful import Resource, Api, reqparse, fields,
marshal_with

app = Flask('__main__')
api = Api(app)

parser = reqparse.RequestParser()
parser.add_argument("first_name")
parser.add_argument("last_name")
parser.add_argument("role")
parser.add_argument("salary", type=int, help='Salary must be an
integer')
out_fields_1 = {"first_name": fields.String,"role": fields.String}
out_fields_2 = {"first_name": fields.String,"last_name":
fields.String}
out_fields_3 = {"first_name": fields.String, "salary":
fields.Integer}

class MyApi(Resource):
    @marshal_with(out_fields_2)
    def get(self):
        info = parser.parse_args()
        return info

    @marshal_with(out_fields_1)
    def post(self):
        info = parser.parse_args()
        return info

    @marshal_with(out_fields_3)
    def put(self):
        info = parser.parse_args()
        return info

api.add_resource(MyApi, '/myinfo')
app.run(debug = True)
```

Sub questions

Question Number : 225 Question Id : 640653356121 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

If the flask application is running locally on URL "http://127.0.0.1:5000/myinfo", what will be the output of the following curl request?

```
curl "http://127.0.0.1:5000/myinfo" -X POST -d "{\"first_name\":
\"Shrivatsa\",\"last_name\":\"Tandon\",\"role\":\"Analyst\",\"salary\
": 50000}" -H "Content-Type: application/json"
```

Options :

6406531180642. ✖

```
{
    "first_name": "Shrivatsa",
    "last_name": "Tandon"
}
```

6406531180643. ✖

```
{
    "first_name": "Shrivatsa",
    "salary": 50000
}
```

6406531180644. ✔

```
{
    "first_name": "Shrivatsa",
    "role": "Analyst"
}
```

6406531180645. ✖

```
{
    "first_name": "Shrivatsa",
    "last_name": "Tandon",
    "role": "Analyst",
    "salary": 50000
}
```

Question Number : 226 Question Id : 640653356122 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

If the flask application is running locally on URL "http://127.0.0.1:5000/myinfo", what will be the output of the following Python code snippet?

```
import requests

data = {"first_name":"Rajnish",
        "last_name":"Dey",
        "role":"Manager",
        "salary":"10 thousand"}

response = requests.put('http://127.0.0.1:5000/myinfo', data = data)
print(response.json())
```

Options :

6406531180646. ✖
```
{
  'first_name': 'Rajnish',
  'salary': '10 thousand'
}
```

6406531180647. ✖
```
{
   'first_name': 'Rajnish',
   'role': 'Manager'
}
```

6406531180648. ✖
```
{
  'first_name': 'Rajnish',
  'last_name': 'Dey'
}
```

6406531180649. ✔ None of these