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 :	64065341234
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 :	64065387444
Question Shuffling Allowed :	No
Is Section Default? :	null

Question Number : 105 Question Id : 640653608764 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 :

6406532033473. ✔ YES

6406532033474. ✖ NO

Sub-Section Number :	2
Sub-Section Id :	64065387445
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 106 Question Id : 640653608765 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 three components of a todo list app based on an MVC architecture.

A - stores a list of all the tasks that the user has submitted along with attributes like completion status.
B - can display which tasks were completed, which tasks are pending, receive new tasks from the user etc.
C - can set a task in the list of tasks as completed, add a new task to the list of tasks, delete old tasks, retrieve a list of pending tasks and so on.

Given the above capabilities of the different components, choose the correct option:

Options :

6406532033475. ✔ model - A, view - B, controller - C

6406532033476. ✖ model - B, view - A, controller - C

6406532033477. ✖ model - C, view - B, controller - A

6406532033478. ✖ model - A, view - C, controller - B

Question Number : 107 Question Id : 640653608766 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 statements is/are correct?

Statement 1: In client-server architecture, the client and the server MUST be on separate machines

Statement 2: In peer-to-peer architecture, the network is always more fault-tolerant

Options :

6406532033479. ✖ Both statement 1 and statement 2 are correct.

6406532033480. ✖ Statement 1 is correct but statement 2 is incorrect.

6406532033481. ✖ Statement 2 is correct but statement 1 is incorrect.

6406532033482. ✔ Neither statement 1 nor statement 2 is correct.

Question Number : 108 Question Id : 640653608773 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

In relational databases, a column stores (1) _____ and a row stores (2) _____.

Options :

6406532033507. ✖ a field, multiple entries

6406532033508. ✔ a field, a single entry

6406532033509. ✖ a single entry, a field

6406532033510. ✖ multiple entries, a field

Question Number : 109 Question Id : 640653608777 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 helps us to create custom HTML elements?

Options :

6406532033523. ✖ SVG

6406532033524. ✔ Web Components

6406532033525. ✖ Web API

6406532033526. ✖ None of these

Question Number : 110 Question Id : 640653608781 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 :

6406532033539. ✖ `git rm <filename>`

6406532033540. ✔ `git rm --cached <filename>`

6406532033541. ✖ `git del <filename>`

6406532033542. ✖ `git del --cached <filename>`

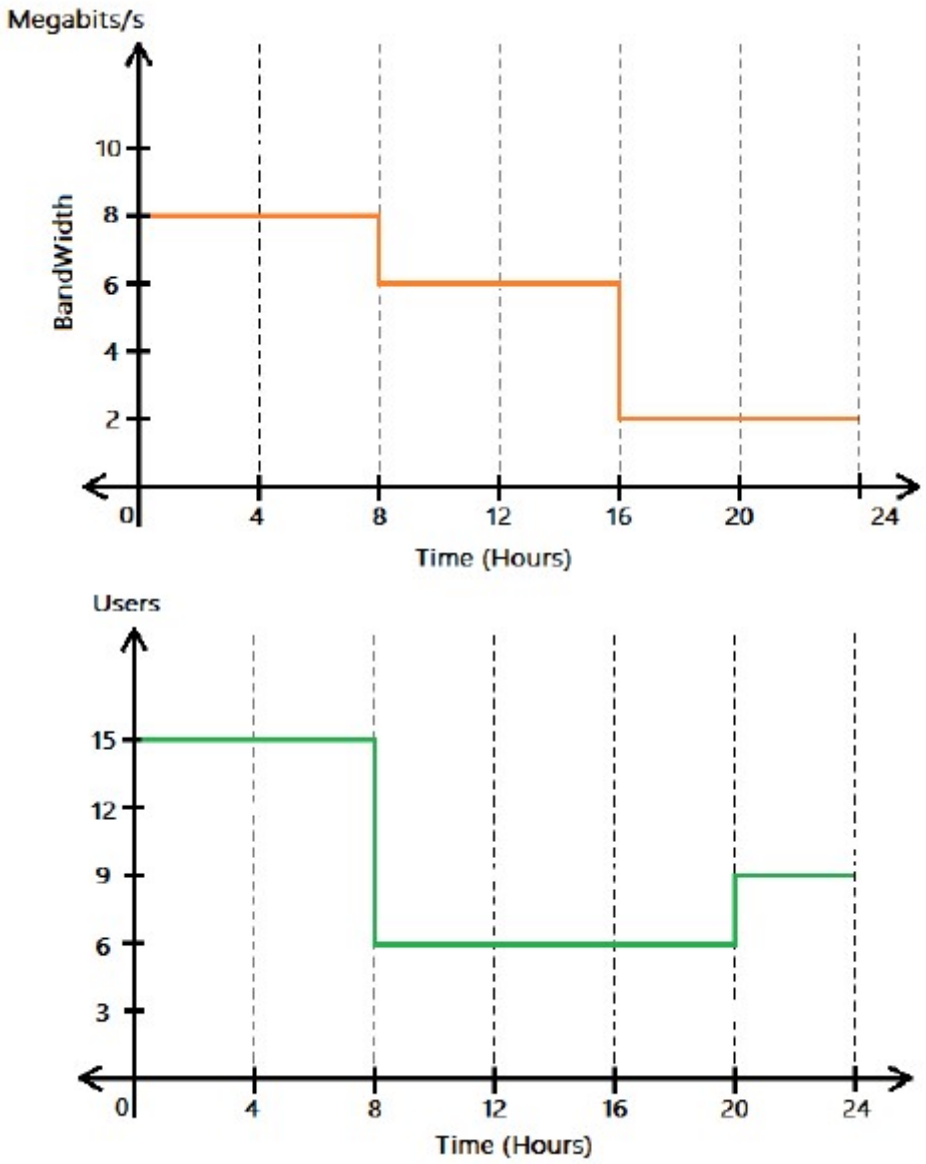
Sub-Section Number :	3
Sub-Section Id :	64065387446
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 111 Question Id : 640653608767 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 and number of users connected to a network for an entire day (24 hours). What will be the total data consumed (in Gigabytes) by the user that is connected to the internet network throughout the day?



Options :

6406532033483. ✖ 5.216

6406532033484. ✖ 52.16

6406532033485. ✔ 6.52

6406532033486. ✖ 65.20

Sub-Section Number :	4
Sub-Section Id :	64065387447
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 112 Question Id : 640653608768 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 a client ‘C’ and a server ‘S’, separated by distance ‘D’ are connected by a fictitious medium in which the speed of light is ‘v’ m/sec. If ‘N’ is the number of consecutive requests that can made in a second by the client ‘C’ (i.e A new request can be made only after receiving the response from the previous request.), Which of the following changes would double the number ‘N’?

Options :

6406532033487. ✖ A change of medium where the speed of light is v/2 m/sec.

6406532033488. ✔ A change of medium where the speed of light is 2v m/sec.

6406532033489. ✔ Reduce the distance between C and S from D to D/2.

6406532033490. ✖ Increase the distance between C and S from D to 2D.

Question Number : 113 Question Id : 640653608774 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 flask app.

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

@app.route('/home/<path:directory>')
def find_course(directory):
    if "ML" in directory:
        return f"Welcome to online course on Data Science!"
    else:
        abort(404)

@app.errorhandler(404)
def page_not_found(error):
    return "<h2>Sorry, No course found!<!h2>"

app.run()
```

If the application is running locally on http://127.0.0.1:5000, select the correct options.

Options :

6406532033511. ✔

For the URL:

`http://127.0.0.1:5000/home/course/data_science?course=ML`

The browser will render:

Sorry, No course found!

6406532033512. ✔

For the URL:

`http://127.0.0.1:5000/home/course/data_science_ML?course=DL`

The browser will render:

Welcome to online course on Data Science!

6406532033513. ✖

For the URL:

`http://127.0.0.1:5000/home/course/programming?course=DBMS`

The browser will render:

Welcome to online course on Programming!

For the URL:

```
http://127.0.0.1:5000/home/course/DBMS?course=Data_structures
```

The browser will render:

Sorry, No course found!

6406532033514. ✖

Question Number : 114 Question Id : 640653608779 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

Which of the following is/are a correct way to use for loop in jinja?

Options :

6406532033531. ✔

```
from jinja2 import Template
t=Template("numbers divisible by 2: {% for n in range (0,10,2) %}
{{n}}""{% endfor %}")
print(t.render())
```

6406532033532. ✔

```
<ul>
{% for item in seq %}
    <li>{{ item }}</li>
{% endfor %}
</ul>
```

6406532033533. ✖

```
from jinja2 import Template
t=Template("My favourite numbers are: {% for n in [2,4,6] {{n}},""{%
endfor %}")
print(t.render())
```

6406532033534. ✔

```
from jinja2 import Template
my_statement = Template("The special series is:{% for n in
range(1,15) -%}   {{n%3}}" "{%- endfor %}")
out = my_statement.render()
print(out)
```

Question Number : 115 Question Id : 640653608793 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 :

6406532033584. ✖

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

6406532033585. ✖

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

6406532033586. ✔ `api.add_resource(TodoSimple, "/api/<int:todo_id>", "/api")`

6406532033587. ✔ `api.add_resource(TodoSimple, "/<int:todo_id>", "/api")`

Sub-Section Number :	5
Sub-Section Id :	64065387448
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 116 Question Id : 640653608769 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 HTML Document.

```
<!DOCTYPE html>
<head>
    <title>Test Document</title>
    <style>
        h4, span {
            display: inline-block;
            width: 200px;
        }
    </style>
</head>
<body>
    <h4>Statement 1 from Document.</h4>
    <h4>Statement 2 from Document.</h4>
    <span>Statement 3 from Document.</span>
    <span>Statement 4 from Document.</span>
</body>
</html>
```

How will the browser render the HTML document given above?

Options :

6406532033491. ✖

Statement 1 from Document.

Statement 2 from Document.

Statement 3 from Document. Statement 4 from Document.

6406532033492. ✖

Statement 1 from Document. Statement 2 from Document.

Statement 3 from Document.

Statement 4 from Document.

6406532033493. ✖

Statement 1 from Document.

Statement 2 from Document.

Statement 3 from Document.

Statement 4 from Document.

6406532033494. ✔

Statement 1 from Document. **Statement 2 from Document.** Statement 3 from Document. Statement 4 from Document.

Question Number : 117 Question Id : 640653608770 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 two numbers N1 = 113_a and N2 = 365_b where a and b are the bases. If the decimal equivalent of N1 is numerically greater than the decimal equivalent of N2, then the possible values of a and b would be.

Options :

6406532033495. ✖ a = 10, b = 8

6406532033496. ✖ a = 8, b = 10

6406532033497. ✔ a = 16, b = 8

6406532033498. ✖ a = 16, b = 10

Question Number : 118 Question Id : 640653608771 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 app and Jinja2 template.

app.py

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

@app.route('/')
def index():
  return render_template("index.html", links=['hoME', 'PROfile',
'Contact', 'SITEMAP'])

app.run()
```

index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Macro</title>
</head>
<body>
    {% macro unordered_list(items)%}
        <ul>
        {% for item in items %}
            <li><a href="/{{item}}">{{item|capitalize}}</a></li>
        {% endfor %}
        </ul>
    {% endmacro %}
    {{ unordered_list(links) }}
</body>
</html>
```

If the flask app is running locally on `http://127.0.0.1:5000`. What will be the output on the browser for the base URL?

Options :

6406532033499. ✔

- Home
- Profile
- Contact
- Sitemap

6406532033500. ✖

- home
- profile
- contact
- sitemap

6406532033501. ✖

- HoME
- PROfile
- Contact
- SITEMAP

6406532033502. ✖

Question Number : 119 Question Id : 640653608775 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.

file.py

```
import sys

courses = {
           1: "App Dev I",
           2: "App Dev II",
           3: "Machine Learning",
           4: "Deep learning"
          }

arguments = sys.argv

if courses[int(sys.argv[2])] in "App Dev II":
    print("course found",courses[int(sys.argv[2])])
else:
    print("No course found!")
```

What will be output on the terminal for the command `python file.py courses 1`?

Options :

6406532033515. ✖ `course found App Dev II`

6406532033516. ✔

```
course found App Dev I
```

6406532033517. ✖

```
No course found!
```

6406532033518. ✖

```
IndexError: list index out of range
```

Question Number : 120 Question Id : 640653608776 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.

```
def sample_function(x):
    ser = []
    for i in range(x):
        ser.append((i)**2)
    return ser

def test_func1():
    assert 36 in sample_function(7)

def test_func2():
    assert 64 in sample_function(8)

def sample_test3():
    assert 81 in sample_function(11)
```

What will be the summary of the output for the command `pytest file.py` on the terminal?

Options :

6406532033519. ✖

```
=============== 1 failed, 2 passed in 0.30s ================
```

6406532033520. ✖

```
=============== 2 failed, 1 passed in 0.30s ================
```

6406532033521. ✖

```
==================== 3 failed in 0.30s =====================
```

6406532033522. ✔

```
=============== 1 failed, 1 passed in 0.30s ================
```

Question Number : 121 Question Id : 640653608778 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.

log.py

```
import logging
import sys

logging.basicConfig(level=logging.WARNING,
                    format='%(asctime)s - %(levelname)s - %(message)s')

def check_val(value):
    if value < 0:
        raise ValueError("Invalid value: Please enter a positive value.")
    else:
        logging.info("Value added: %s", value)

try:
    input_value = int(sys.argv[1])
    check_val(input_value)
except ValueError as ve:
    logging.exception("Exception occurred: %s", str(ve))
```

What will be the output on the terminal for the command: `python log.py 34` ?

Options :

6406532033527. ✖

```
2023-08-14 21:01:05,684 - INFO - Value added: 34
```

6406532033528. ✖

```
2023-08-14 21:01:05,684 - WARNING - Value added: 34
```

6406532033529. ✖

```
ValueError: Invalid value: Please enter a positive value.
```

6406532033530. ✔ None.

Question Number : 122 Question Id : 640653608782 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 :

6406532033543. ✖ 1-d, 2-a, 3-b, 4-c

6406532033544. ✖ 1-c, 2-d, 3-a, 4-b

6406532033545. ✔ 1-c, 2-d, 3-b, 4-a

6406532033546. ✖ 1-b, 2-d, 3-c, 4-a

Question Number : 123 Question Id : 640653608783 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 :

6406532033547. ✖ A → 1, B → 2, C → 3, D → 4

6406532033548. ✖ A → 4, B → 3, C → 2, D → 1

6406532033549. ✔ A → 4, B → 1, C → 2, D → 3

6406532033550. ✖ A → 3, B → 2, C → 1, D → 4

Question Number : 124 Question Id : 640653608786 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 :

6406532033556. ✖

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

6406532033557. ✖

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

6406532033558. ✔

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

6406532033559. ✖

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

Question Number : 125 Question Id : 640653608788 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 :

6406532033564. ✖ **Welcome to MAD I**

6406532033565. ✔ **IITM BS Degree**

Welcome to MAD I

6406532033566. ✖ **Welcome to MAD I**

IITM BS Degree

6406532033567. ✖ Error

Question Number : 126 Question Id : 640653608795 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 :

6406532033592. ✖

```
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())
```

6406532033593. ✖

```
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())
```

6406532033594. ✔

```
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())
```

6406532033595. ✖

```
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))
```

Sub-Section Number : 6

Sub-Section Id : 64065387449

Question Shuffling Allowed : Yes

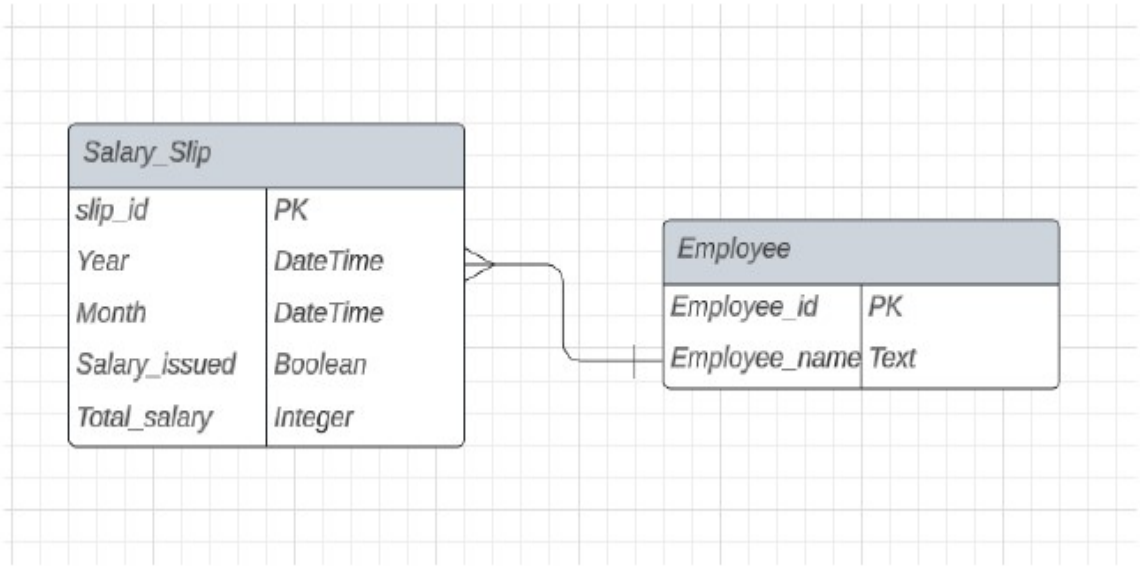
Is Section Default? : null

Question Number : 127 Question Id : 640653608772 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

What can be inferred from the Entity-Relationship Diagram below:



Options :

6406532033503. ✔ An employee can exist without having any salary slips

6406532033504. ✖ A salary slip can exist without belonging to any employee

6406532033505.

✖ An employee needs to have at least one salary slip

6406532033506. ✔ A salary slip must belong to one and only one employee

Question Number : 128 Question Id : 640653608789 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 :

6406532033568. ✔ The AI voice

6406532033569. ✔ The LED light around the device

6406532033570. ✖ The body of the device

6406532033571. ✖ None of these

Question Number : 129 Question Id : 640653608790 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 :

6406532033572. ✖ They have high cost and low durability.

6406532033573. ✔ They have low cost and high durability.

6406532033574. ✖ Latency of retrieval is very low.

6406532033575. ✔ Latency of retrieval is very high.

Sub-Section Number : 7

Sub-Section Id : 64065387450

Question Shuffling Allowed : Yes

Is Section Default? : null

Question Number : 130 Question Id : 640653608780 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

Consider the following flask application.

app.py

```
from flask import Flask, render_template, request
app = Flask(__name__)
@app.route('/', methods=['GET'])
def my_form():
    if request.method == 'GET':
        if(request.args.get('name') == None):
            return """
                      <form method="GET" action ="/">
                         <label for="name">Enter a name: </label>
                         <input type="text" name="name" id="name"></input>
                         <input type="submit" name="btnnum"
                            id="btnnum"></input>
                      </form>
                   """
        elif(request.args.get('name') == ''):
            return "<h1>Invalid name</h1>"
        else:
            name = request.args.get('name')
            return f"<h2>My name is {name}</h2>"

app.run()
```

If this flask app is running locally on http://localhost:5000, then which of the following statements is/are incorrect?

Options :

For http://127.0.0.1:5000/?name=Will+Smith&btnnum=Submit the rendered output is:

6406532033535. ✖

My name is Will Smith

If the form rendered at http://127.0.0.1:5000/ is submitted with value "Will Smith", then the rendered output is:

6406532033536. ✔

My name is Will

6406532033537. ✔ For the URL http://127.0.0.1:5000/?name, the app will redirect to the base URL.

6406532033538. ✔ If the form rendered at base URL is submitted with all the fields empty, then it will throw "Not Found" error.

Question Number : 131 Question Id : 640653608796 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 :

6406532033596. ✔

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

6406532033597. ✔

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

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

6406532033598. ✖

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

6406532033599. ✖

Sub-Section Number : 8

Sub-Section Id : 64065387451

Question Shuffling Allowed : Yes

Is Section Default? : null

Question Number : 132 Question Id : 640653608784 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=121

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

121121

Sub-Section Number :	9
Sub-Section Id :	64065387452
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 133 Question Id : 640653608785 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 :

6406532033552. ✖

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

6406532033553. ✔

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

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

6406532033554. ✖

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

6406532033555. ✖

Question Number : 134 Question Id : 640653608787 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 :

6406532033560. ✖ `== 2 failed, 1 passed in 0.17s ===`

6406532033561. ✔ `== 1 failed, 1 passed, 1 deselected in 0.17s ===`

6406532033562. ✖ `== 2 failed, 1 deselected in 0.17s ===`

6406532033563. ✖ `== 1 failed, 2 passed in 0.17s ===`

Question Number : 135 Question Id : 640653608791 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 :

6406532033576. ✖ `('hello world!', 'hello world!', 'hello world!', 'hello world!', 'hello world!')`

6406532033577. ✖ `(5, 5, 5, 5, 5)`

6406532033578. ✖ `25`

6406532033579. ✔ `(5, 'hello world!', 5, 'hello world!', 5, 'hello world!', 5, 'hello world!', 5, 'hello world!')`

Question Number : 136 Question Id : 640653608792 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 :

6406532033580. ✖ a-2, b-1, c-5, d-6

6406532033581. ✖ a-5, b-1, c-2, d-6

6406532033582. ✔ a-5, b-3, c-2, d-4

6406532033583.

✖ a-2, b-3, c-4, d-6

Question Number : 137 Question Id : 640653608794 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 :

6406532033588. ✖ $O(logN)$

6406532033589. ✖ $O(N)$

6406532033590. ✖ $O(NlogN)$

6406532033591. ✔ $O(N^2)$

TDS

Section Id :	64065341235
Section Number :	7
Section type :	Online
Mandatory or Optional :	Mandatory
Number of Questions :	62