

# CSI3023

## Advanced Server-Side Programming

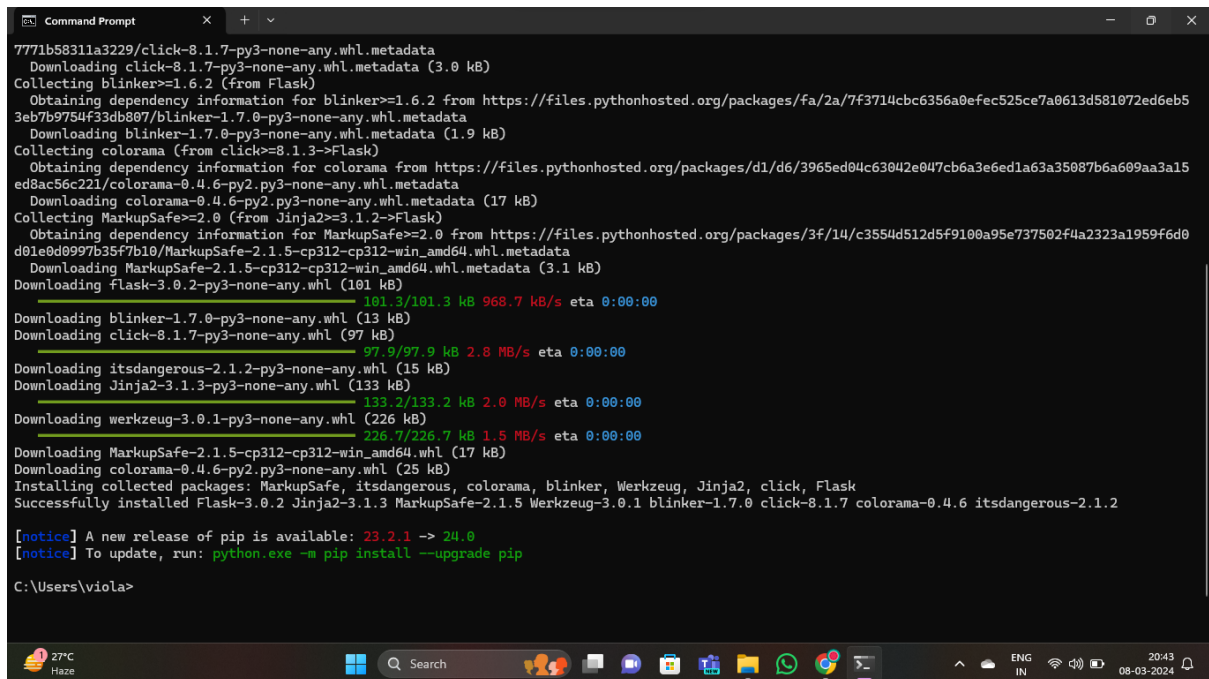
Winter Semester 2023 – '24

Digital Assignment - 4

**SARAGA S**  
20MIC0081

## Create an application implementing a RESTful API:

pip install Flask



```
7771b58311a3229/click-8.1.7-py3-none-any.whl.metadata
Downloading click-8.1.7-py3-none-any.whl.metadata (3.0 kB)
Collecting blinker>=1.6.2 (from Flask)
  Obtaining dependency information for blinker>=1.6.2 from https://files.pythonhosted.org/packages/fa/2a/7f3714cbc6356a0efec525ce7a0613d581072ed6eb53eb7b9754f33db8077/blinker-1.7.0-py3-none-any.whl.metadata
  Downloading blinker-1.7.0-py3-none-any.whl.metadata (1.9 kB)
Collecting colorama (from click>=8.1.3->Flask)
  Obtaining dependency information for colorama from https://files.pythonhosted.org/packages/d1/d6/3965ed04c63042e047cb6a3e6ed1a63a35087b6a609aa3a15ed8ac56c221/colorama-0.4.6-py2.py3-none-any.whl.metadata
  Downloading colorama-0.4.6-py2.py3-none-any.whl.metadata (17 kB)
Collecting MarkupSafe>=2.0 (from Jinja2>=3.1.2->Flask)
  Obtaining dependency information for MarkupSafe>=2.0 from https://files.pythonhosted.org/packages/3f/14/c3554d512d5f9100a95e737502f4a2323a1959f6d0d01e0d0997b35f7b10/MarkupSafe-2.1.5-cp312-cp312-win_amd64.whl.metadata
  Downloading MarkupSafe-2.1.5-cp312-cp312-win_amd64.whl.metadata (3.1 kB)
Downloading flask-3.0.2-py3-none-any.whl (101 kB)
101.3/101.3 kB 968.7 kB/s eta 0:00:00
Downloading blinker-1.7.0-py3-none-any.whl (13 kB)
Downloading click-8.1.7-py3-none-any.whl (97 kB)
97.9/97.9 kB 2.8 MB/s eta 0:00:00
Downloading itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Downloading Jinja2-3.1.3-py3-none-any.whl (133 kB)
133.2/133.2 kB 2.0 MB/s eta 0:00:00
Downloading werkzeug-3.0.1-py3-none-any.whl (226 kB)
226.7/226.7 kB 1.5 MB/s eta 0:00:00
Downloading MarkupSafe-2.1.5-cp312-cp312-win_amd64.whl (17 kB)
Downloading colorama-0.4.6-py2.py3-none-any.whl (25 kB)
Installing collected packages: MarkupSafe, itsdangerous, colorama, blinker, Werkzeug, Jinja2, click, Flask
Successfully installed Flask-3.0.2 Jinja2-3.1.3 MarkupSafe-2.1.5 Werkzeug-3.0.1 blinker-1.7.0 click-8.1.7 colorama-0.4.6 itsdangerous-2.1.2

[notice] A new release of pip is available: 23.2.1 -> 24.0
[notice] To update, run: python.exe -m pip install --upgrade pip

C:\Users\viola>
```

## CODE:

```
from flask import Flask, jsonify, request

app = Flask(__name__)

# Sample data (you can replace this with a database)
tasks = [
    {
        'id': 1,
        'title': 'Task 1',
        'done': False
    },
    {
        'id': 2,
        'title': 'Task 2',
        'done': False
    }
]
```

```

# Get all tasks
@app.route('/tasks', methods=['GET'])
def get_tasks():
    return jsonify({'tasks': tasks})

# Get a specific task by ID
@app.route('/tasks/<int:task_id>', methods=['GET'])
def get_task(task_id):
    task = next((task for task in tasks if task['id'] == task_id), None)
    if task is None:
        return jsonify({'error': 'Task not found'}), 404
    return jsonify({'task': task})

# Create a new task
@app.route('/tasks', methods=['POST'])
def create_task():
    if not request.json or 'title' not in request.json:
        return jsonify({'error': 'Invalid data'}), 400

    new_task = {
        'id': tasks[-1]['id'] + 1,
        'title': request.json['title'],
        'done': False
    }
    tasks.append(new_task)

    return jsonify({'task': new_task}), 201

# Update an existing task
@app.route('/tasks/<int:task_id>', methods=['PUT'])
def update_task(task_id):
    task = next((task for task in tasks if task['id'] == task_id), None)
    if task is None:
        return jsonify({'error': 'Task not found'}), 404

    if 'title' in request.json:
        task['title'] = request.json['title']
    if 'done' in request.json:
        task['done'] = request.json['done']

    return jsonify({'task': task})

# Delete a task
@app.route('/tasks/<int:task_id>', methods=['DELETE'])
def delete_task(task_id):
    task = next((task for task in tasks if task['id'] == task_id), None)
    if task is None:
        return jsonify({'error': 'Task not found'}), 404

```

```
tasks.remove(task)
return jsonify({'result': True})

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

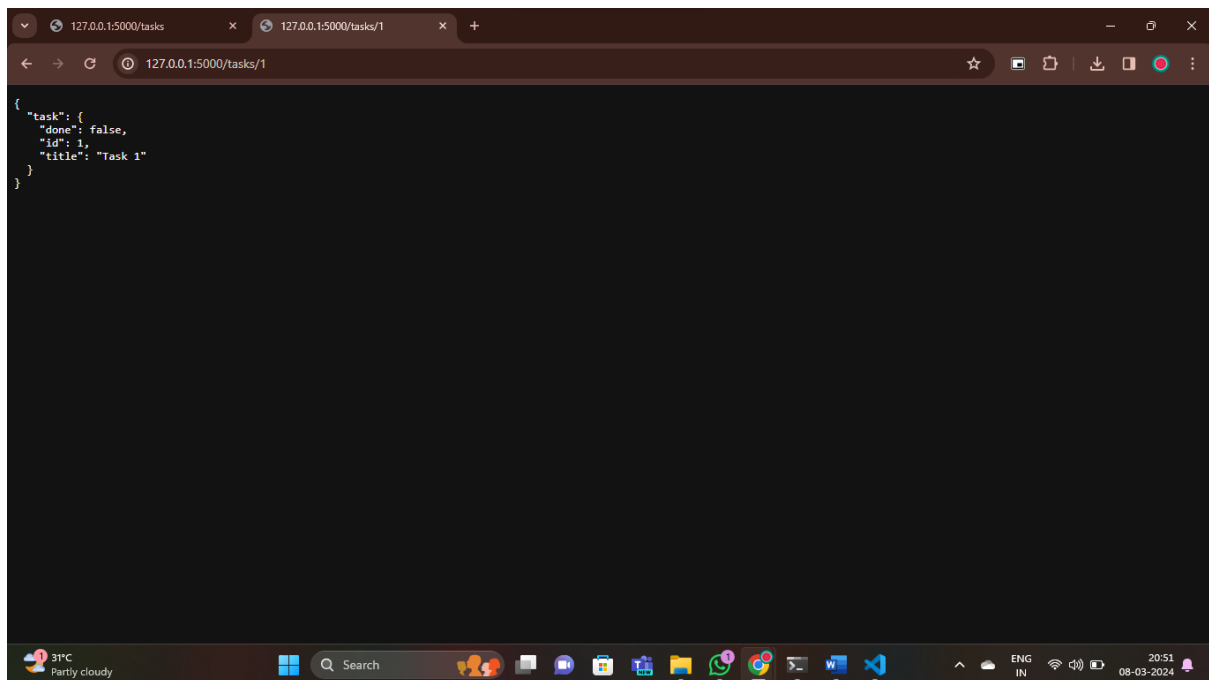
## OUTPUT:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\viola\Downloads\ADV SEV> python app.py
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 970-528-806
```

```
{
  "tasks": [
    {
      "done": false,
      "id": 1,
      "title": "Task 1"
    },
    {
      "done": false,
      "id": 2,
      "title": "Task 2"
    }
  ]
}
```



To get the list of tasks (GET request):

`Invoke-RestMethod -Uri http://127.0.0.1:5000/tasks -Method Get`

To create a new task (POST request):

`Invoke-RestMethod -Uri http://127.0.0.1:5000/tasks -Method Post -Body '{"title": "New Task"}' -ContentType 'application/json'`

To update an existing task by ID (replace 1 with the desired task ID) (PUT request), run:

`Invoke-RestMethod -Uri http://127.0.0.1:5000/tasks/1 -Method Put -Body '{"title": "Updated Task", "done": true}' -ContentType 'application/json'`

To get the updated list of tasks, run the GET request again:

`Invoke-RestMethod -Uri http://127.0.0.1:5000/tasks -Method Get`

To delete a task by ID (replace 1 with the desired task ID) (DELETE request), run:

`Invoke-RestMethod -Uri http://127.0.0.1:5000/tasks/1 -Method Delete`

## OUTPUT:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\viola\Downloads\ADV SEV> Invoke-RestMethod -Uri http://127.0.0.1:5000/tasks -Method Get

tasks
-----
@{done=False; id=1; title=Task 1}, @{{done=False; id=2; title=Task 2}}

PS C:\Users\viola\Downloads\ADV SEV> Invoke-RestMethod -Uri http://127.0.0.1:5000/tasks -Method Post -Body '{"title": "New Task"}' -ContentType 'application/json'

task
----
@{done=False; id=3; title=New Task}

PS C:\Users\viola\Downloads\ADV SEV> Invoke-RestMethod -Uri http://127.0.0.1:5000/tasks/1 -Method Put -Body '{"title": "Updated Task", "done": true}' -ContentType 'application/json'

task
----
@{done=True; id=1; title=Updated Task}

PS C:\Users\viola\Downloads\ADV SEV> Invoke-RestMethod -Uri http://127.0.0.1:5000/tasks -Method Get

tasks
-----
@{done=True; id=1; title=Updated Task}, @{{done=False; id=2; title=Task 2}, @{{done=False; id=3; title=New Task}}

31°C Partly cloudy 20:58 08-03-2024
```

```
-----
@{done=False; id=1; title=Task 1}, @{{done=False; id=2; title=Task 2}}

PS C:\Users\viola\Downloads\ADV SEV> Invoke-RestMethod -Uri http://127.0.0.1:5000/tasks -Method Post -Body '{"title": "New Task"}' -ContentType 'application/json'

task
----
@{done=False; id=3; title=New Task}

PS C:\Users\viola\Downloads\ADV SEV> Invoke-RestMethod -Uri http://127.0.0.1:5000/tasks/1 -Method Put -Body '{"title": "Updated Task", "done": true}' -ContentType 'application/json'

task
----
@{done=True; id=1; title=Updated Task}

PS C:\Users\viola\Downloads\ADV SEV> Invoke-RestMethod -Uri http://127.0.0.1:5000/tasks -Method Get

tasks
-----
@{done=True; id=1; title=Updated Task}, @{{done=False; id=2; title=Task 2}, @{{done=False; id=3; title=New Task}}

PS C:\Users\viola\Downloads\ADV SEV> Invoke-RestMethod -Uri http://127.0.0.1:5000/tasks/1 -Method Delete

result
-----
True

PS C:\Users\viola\Downloads\ADV SEV>
```