# List of tasks with priority

tasks = [

("Task 1", 3),

("Task 2", 1),

("Task 3", 4),

("Task 4", 2),

("Task 5", 5)

]

# Sort the tasks by priority (second element of the tuple)

sorted\_tasks = sorted(tasks, key=lambda x: x[1])

# Print sorted tasks

print("Sorted tasks by priority:")

for task in sorted\_tasks:

print(f"Task: {task[0]}, Priority: {task[1]}")

In this example, the sorted() function sorts the list of tuples based on the second element of each tuple (the priority) by specifying key=lambda x: x[1]. The sorted tasks are then printed in order of their priority.

# List of tasks

tasks = [("Task 1", 3), ("Task 2", 1), ("Task 3", 4)]

# New task to append

new\_task = ("Task 4", 2)

# Append the new task to the list

tasks.append(new\_task)

# Print the updated list

print("Updated list of tasks:")

for task in tasks:

print(f"Task: {task[0]}, Priority: {task[1]}")

In this example, the new task tuple ("Task 4", 2) is appended to the existing tasks list using the append() method. The updated list of tasks is then printed.

# Function to append a task tuple to the list

def add\_task(tasks\_list, task, priority):

task\_tuple = (task, priority)

tasks\_list.append(task\_tuple)

# Example usage

if \_\_name\_\_ == "\_\_main\_\_":

tasks = [("Task 1", 3), ("Task 2", 1), ("Task 3", 4)]

# New task to add

new\_task = "Task 4"

new\_priority = 2

# Add the new task using the function

add\_task(tasks, new\_task, new\_priority)

# Print the updated list

print("Updated list of tasks:")

for task in tasks:

print(f"Task: {task[0]}, Priority: {task[1]}")

---------------------------------------------------------------------------------

# Function to append a task tuple to the list

def add\_task(tasks\_list, task, priority):

task\_tuple = (task, priority)

tasks\_list.append(task\_tuple)

# Function to remove a task tuple from the list

def remove\_task(tasks\_list, task):

for t in tasks\_list:

if t[0] == task:

tasks\_list.remove(t)

return

print(f"Task {task} not found.")

# Example usage

if \_\_name\_\_ == "\_\_main\_\_":

tasks = [("Task 1", 3), ("Task 2", 1), ("Task 3", 4)]

# New task to add

new\_task = "Task 4"

new\_priority = 2

# Add the new task using the function

add\_task(tasks, new\_task, new\_priority)

# Remove a task using the function

remove\_task(tasks, "Task 2")

# Print the updated list

print("Updated list of tasks:")

for task in tasks:

print(f"Task: {task[0]}, Priority: {task[1]}")

---------------------------------------------------------------------------------------