**SOURCE CODE:**

*import sched*

*import time*

*import threading*

*class RealTimeTask:*

*def \_\_init\_\_(self, name, priority, execution\_time):*

*self.name = name*

*self.priority = priority*

*self.execution\_time = execution\_time*

*def real\_time\_scheduler(tasks):*

*s = sched.scheduler(time.time, time.sleep)*

*for task in tasks:*

*s.enter(task.priority, 1, execute\_real\_time\_task, argument=(task,))*

*print("Real-time task execution order:")*

*s.run()*

*def execute\_real\_time\_task(task):*

*print(f"Executing task: {task.name}, Priority: {task.priority}, Execution Time: {task.execution\_time} seconds")*

*time.sleep(task.execution\_time)*

*print(f"Task {task.name} completed")*

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

*# Define real-time tasks with name, priority, and execution time*

*task1 = RealTimeTask("Task1", 1, 3)*

*task2 = RealTimeTask("Task2", 2, 2)*

*task3 = RealTimeTask("Task3", 3, 1)*

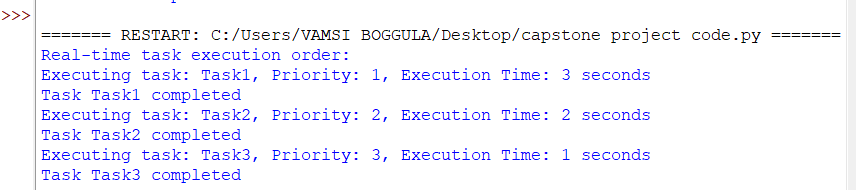
*# List of real-time tasks*

*tasks\_list = [task1, task2, task3]*

*# Start real-time scheduler*

*real\_time\_scheduler(tasks\_list)*

**Output:**

****