```
import threading
import time
from datetime import datetime
from robodk import robolink
import csv

# Connect with ROBODK
RDK = robolink.Robolink()

# Hardware setup
KUKA = RDK.Item('KUKA KR 30 HA')
if not KUKA.Valid():
    raise Exception("Robot not selected or not available")

# File path for the CSV
file_path = r'C:\Users\XXX\Desktop\Position4.csv'

# Time interval for monitoring in seconds
interval = 0.01


def monitor_position(robot, interval, file_path):
    with open(file_path, 'w', newline='') as file:  # Open file in
write mode
        csv_writer = csv.writer(file)
        csv_writer.writerow(["Timestamp", "Position"])  # Write
header
        #csv_writer.writerow(["Timestamp"])  # Write header
        try:
            while True:
                #pos = robot.Pose()
                pos = robot.Joints().list()
                current_time = datetime.now().strftime("%Y-%m-%d %H:
%M:%S.%f")
                #csv_writer.writerow([current_time])
                if robot.Connect() > 0:
                    print("Connected")
                print(f"Timestamp: {current_time} | Current
Position: {pos}")
                csv_writer.writerow([current_time, pos])
                file.flush()  # Ensure data is written to file
                time.sleep(interval)
        except KeyboardInterrupt:
            print("Stopped position tracking")


thread = threading.Thread(target=monitor_position, args=(KUKA,
interval, file_path))
thread.start()


def run_program(program_name):
    program = RDK.Item(program_name, robolink.ITEM_TYPE_PROGRAM)
    if not program.Valid():
```

```
        raise Exception(f"Program {program_name} not found")
    program.RunProgram()

#run_program('Scanning') Commented out, not relevant for just
tracking the position
```