In [21]:
```
from langchain_experimental.tools import PythonREPLTool
from langchain.python import PythonREPL
from langchain_experimental.agents.agent_toolkits import create_python_ag
```

In [22]:
```
from langchain.agents import initialize_agent, create_react_agent
from langchain.agents import AgentType
from langchain.chat_models import ChatOpenAI
from langchain.agents import tool
import os
```

In [5]:
```
# os.environ['OPENAI_API_KEY'] = "##"
```

In [23]:
```
llm = ChatOpenAI(model= "gpt-4o",temperature=0)
```

In [24]:
```
@tool
def create_log_file(file_path):
    ''' This function takes a folder path as String input, an creates a f
    import uuid
    agent_id = str(uuid.uuid4())
    log_file_path = os.path.join(file_path, "log_file.log")


    with open(file_path, 'w') as f:
        f.write("Log file for agent {}\n".format(agent_id))
        success = "created successfully"
    return success
```

In [25]:
```
@tool
def worker_flow(file_path):

    ''' after finishing with leader file, this function takes folder path
    import os
    files_list_file = os.path.join(file_path, "files_list.txt")
    log_file_path = os.path.join(file_path, "log_file.log")

    try:
        if os.path.exists(files_list_file):
            with open(files_list_file, 'r') as f:
                files_to_inspect = json.load(f)

            for infected_file_path in files_to_inspect:
                try:
                    quarantine_path = os.path.join(file_path, os.path.bas
                    os.rename(infected_file_path, quarantine_path)
                    with open(log_file_path, 'a') as log:
                        log.write("Quarantined file: {}, Original path: {
                except Exception as e:
                    with open(log_file_path, 'a') as log:
                        log.write("Error processing file {}: {}\n".format
        feedback = "completed successfully"

                # inspect_file(infected_file_path,log_file_path)
    except Exception as e:
        with open(log_file_path, 'a') as log:
            log.write("Error in worker_flow: {}\n".format(e))
```

```
        feedback ="didn't finish; Terminate"
    return feedback
```

In [26]:
```
@tool
def leader_flow(folder_path, extension):
    ''' this function takes folder_path as String input, which is the pat
    import uuid
    import os
    agent_id = str(uuid.uuid4())
    agent_num_file = os.path.join(folder_path, "agent_num_file.txt")
    files_list_file = os.path.join(folder_path, "files_list.txt")

    with open(agent_num_file, 'w') as f:
        f.write(agent_id)

    infected_files = []
    for root, _, files in os.walk(folder_path):
        for file in files:
            if (file.endswith(extension) ) and ('private' in file or 'key
                infected_files.append(os.path.join(root, file))

    with open(files_list_file, 'w') as f:
        json.dump(infected_files, f)
    feedback = "completed successfully"

    return feedback
```

In [27]:
```
# tools = load_tools( llm=llm)
tools =  [create_log_file,worker_flow,leader_flow]
```

In [28]:
```
agent = initialize_agent(tools,
                         llm,
                         agent=AgentType.STRUCTURED_CHAT_ZERO_SHOT_REACT_
                         verbose=True)
```

In [29]:
```
agent.run("in the following dictionary path '../Project_unit_6/test' and
```

```
> Entering new AgentExecutor chain...
Thought: To quarantine all files with the '.txt' extension in the specifie
d path and its subdirectories, I need to first create a "files_list_file"
that lists all the infected files. Then, I will use the worker_flow to mov
e these files to a quarantine folder.

First, I will use the leader_flow to search for all '.txt' files and creat
e the "files_list_file".

Action:
```
{
  "action": "leader_flow",
  "action_input": {
    "folder_path": "../Project_unit_6/test",
    "extension": ".txt"
  }
}
```

Observation: completed successfully
Thought:To proceed with quarantining the identified '.txt' files, I will n
ow use the `worker_flow` to move these files to a quarantine folder and lo
g the actions taken.

Action:
```
{
  "action": "worker_flow",
  "action_input": {
    "file_path": "../Project_unit_6/test"
  }
}
```

Observation: completed successfully
Thought:It seems that the process of identifying and quarantining the '.tx
t' files in the specified path has been completed successfully. Here is th
e final response:

Action:
```
{
  "action": "Final Answer",
  "action_input": "All '.txt' files in the '../Project_unit_6/test' direct
ory and its subdirectories have been successfully quarantined."
}
```

> Finished chain.
```

Out[29]: "All '.txt' files in the '../Project_unit_6/test' directory and its subd irectories have been successfully quarantined."

In [ ]:

In [ ]: