

```
from typing import List, Optional
```

```
from crewai import Agent as CrewAIAgent
```

```
from crewai import Crew, Process, Task
```

```
from crewai_tools import SerperDevTool
```

```
from loguru import logger
```

```
from swarms import Agent
```

```
class CrewAIAgentWrapper(Agent):
```

```
    """
```

```
    Initialize the CrewAIAgentWrapper.
```

```
    Args:
```

```
        name (str): The name of the agent.
```

```
        role (str): The role of the agent.
```

```
        goal (str): The goal of the agent.
```

```
        backstory (str): The backstory of the agent.
```

```
        tools (Optional[List]): The tools used by the agent (default: None).
```

```
        *args: Variable length argument list.
```

```
        **kwargs: Arbitrary keyword arguments.
```

```
    """
```

```
    def __init__(
```

```
        self,
```

```

name: str,

role: str,

goal: str,

backstory: str,

tools: Optional[List] = None,

*args,

**kwargs,

):

    super().__init__(*args, **kwargs)

    self.name = name

    self.crewai_agent = CrewAIAgent(

        role=role,

        goal=goal,

        backstory=backstory,

        verbose=True,

        allow_delegation=False,

        tools=tools or [],

        *args,

        **kwargs,

    )

```

```

def run(self, task: str, *args, **kwargs):

```

```

    """

```

```

    Run the agent's task.

```

```

    Args:

```

task (str): The task to be performed by the agent.

*args: Variable length argument list.

**kwargs: Arbitrary keyword arguments.

Returns:

Any: The result of the task execution.

"""

try:

```
crew_task = Task(
    description=task,
    agent=self.crewai_agent,
    *args,
    **kwargs,
)

crew = Crew(
    agents=[self.crewai_agent],
    tasks=[crew_task],
    process=Process.sequential,
)
```

```
result = crew.kickoff()
```

```
return result
```

except Exception as e:

```
logger.error(f"An error occurred: {e}")
```

```
return None
```

Usage example

```
search_tool = SerperDevTool()
```

```
crewai_wrapper = CrewAIWrapper(  
    name="ResearchAnalyst",  
    role="Senior Research Analyst",  
    goal="Uncover cutting-edge developments in AI and data science",  
    backstory="""You work at a leading tech think tank.  
Your expertise lies in identifying emerging trends.  
You have a knack for dissecting complex data and presenting actionable insights."""  
    tools=[search_tool],  
)
```

```
result = crewai_wrapper.run(  
    "Analyze the latest trends in quantum computing and summarize the key findings."  
)  
print(result)
```