Building Your Own Crew AI Agent

MIT AI Studio - Homework Assignment

Core Concepts of Crew AI

Agents

Autonomous entities with specific **roles** **goals** , and **backstories** .

Tasks

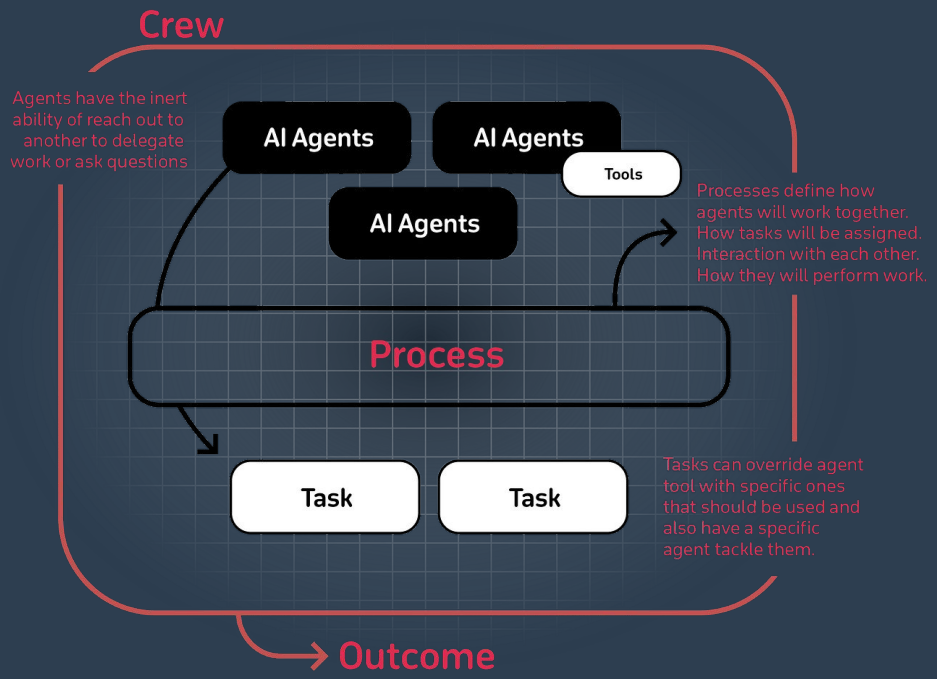
Units of work assigned to agents with clear **descriptions** and **expected outputs** .

Crews

Collections of agents working together through defined **processes** to achieve common objectives.

Tools

External functionalities that extend agent capabilities, such as **search** , **file writing** , or **API access** .



Agents in Crew AI

Defining an Agent

Agents are the core building blocks of any Crew AI system. Each agent has:

Role

Defines the agent's specialty and expertise

Goal

What the agent aims to achieve

Backstory

Provides context and personality

Tools

External capabilities the agent can use

Delegation

Whether the agent can delegate tasks

```
# Example of creating a research agent
from crewai import Agent
from crewai_tools import SerperDevTool

# Create a search tool
search_tool = SerperDevTool()

# Define the research agent
researcher = Agent(
    role="Research Assistant",
    goal="Find accurate and up-to-date information on given topics",
    backstory="""You are a diligent research assistant with
    expertise in academic research and information synthesis.
    You have a keen eye for credible sources and can quickly
    identify the most relevant information for any topic.""",
    verbose=True,
    allow_delegation=False,
    tools=[search_tool]
)
```

Tasks and Crews

Tasks

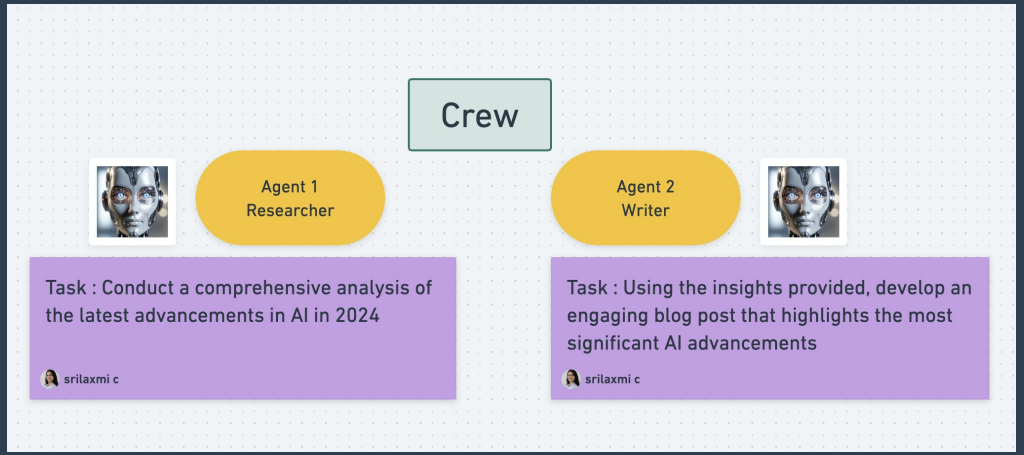
Units of work assigned to agents with specific objectives.

- Detailed **description** of what needs to be done
- Clear **expected output** format
- Assigned to a specific **agent**

Crews

Orchestrates collaboration between multiple agents.

- Manages **agent interactions**
- Defines **workflow** and task sequence
- Facilitates **information sharing**



→ Sequential

Hierarchical

Example Implementation

Research & Writing Crew

A simple example of a two-agent crew that researches a topic and writes an article.

1. Define **agents** with specific roles
2. Create **tasks** with clear descriptions
3. Assemble a **crew** with the agents and tasks
4. Execute the crew with **kickoff()** method
5. Retrieve and use the **results**

```
from crewai import Agent, Task, Crew, Process
from crewai_tools import SerperDevTool, FileWriterTool

# Create agents
researcher = Agent(
    role="Research Assistant",
    goal="Find accurate information on given topics",
    backstory="You are a diligent research assistant with expertise in academic research.",
    verbose=True,
    allow_delegation=False,
    tools=[SerperDevTool()]
)

writer = Agent(
    role="Content Writer",
    goal="Create clear, engaging written content",
    backstory="You are a skilled writer who excels at presenting complex information clearly.",
    verbose=True,
    allow_delegation=False,
    tools=[FileWriterTool()]
)
```

Example Implementation Part 2

Research & Writing Crew

A simple example of a two-agent crew that researches a topic and writes an article.

1. Define **agents** with specific roles
2. Create **tasks** with clear descriptions
3. Assemble a **crew** with the agents and tasks
4. Execute the crew with **kickoff()** method
5. Retrieve and use the **results**

```
# Create tasks
topic = "Artificial Intelligence in Education"

research_task = Task(
    description=f"Research the topic: {topic}. Gather information about key concepts, recent developments, benefits, and challenges.",
    expected_output="A comprehensive research summary with key concepts, developments, benefits, challenges, and sources.",
    agent=researcher
)

writing_task = Task(
    description=f"Write an article about {topic} based on the research findings. Include introduction, key concepts, applications, and conclusion.",
    expected_output="A well-written article saved as 'ai_education_article.md' with proper formatting.",
    agent=writer
)

# Create crew
crew = Crew(
    agents=[researcher, writer],
    tasks=[research_task, writing_task],
    process=Process.sequential,
    verbose=True
)

# Execute the crew
result = crew.kickoff()

# Print the result
print(result)
```

Assignment Requirements

Backend Crew AI Agent

Build a Crew AI system that represents you as an AI Studio student.

- Define at least **two agents** with clear roles, goals, and backstories
- Create at least **two tasks** with descriptions and expected outputs
- Form a **crew** that orchestrates the agents and tasks
- Implement **terminal interaction** to run your agent system
- Write **clean, well-commented code** following Python best practices
- **(Recommended)** Integrate relevant tools to enhance agent capabilities

Submission Checklist

- Python code files for your Crew AI agent
- README with setup and usage instructions
- requirements.txt with dependencies
- All files packaged in a single ZIP archive

Resources and Next Steps

Learning Resources

Official CrewAI GitHub Repository

Source code, examples, and documentation

github.com/crewAIInc/crewAI

DeepLearning.AI Short Course

Multi AI Agent Systems with CrewAI

deeplearning.ai/short-courses/multi-ai-agent-systems-with-crewai

CrewAI Documentation

Comprehensive API reference and tutorials

docs.crewai.com

Video Tutorials

Step-by-step guides for building CrewAI agents

Search "CrewAI Tutorial" on YouTube

Tips for Success

1. **Start Simple, Then Expand**

   Begin with a basic two-agent crew and a clear task. Once that's working, add more complexity with additional agents or tools.

2. **Be Specific with Agent Roles**

   The more detailed your agent `roles`, `goals`, and `backstories`, the better they'll perform their tasks.

3. **Test Incrementally**

   Test each agent and task individually before combining them in a crew to identify and fix issues early.

4. **Leverage Tools**

   Integrate tools like `SerperDevTool` for search, `FileWriterTool` for file operations, or create your own custom tools.

Good luck with your CrewAI agent assignment!