

Swarm

Basic Usage



basic_code.py

```
1 from swarm import Swarm, Agent
2
3 client = Swarm()
4
5 def transfer_to_agent_b():
6     return agent_b
7
8
9 agent_a = Agent(
10     name="Agent A",
11     instructions="You are a helpful agent.",
12     functions=[transfer_to_agent_b],
13 )
14
15 agent_b = Agent(
16     name="Agent B",
17     instructions="Only speak in Haikus.",
18 )
19
20 response = client.run(
21     agent=agent_a,
22     messages=[{"role": "user", "content": "I want to talk to agent
    B."}],
23 )
24
25 print(response.messages[-1]["content"])
```

Output

```
1 Hope glimmers brightly,
2 New paths converge gracefully,
3 What can I assist?
```

Swarm

Instructions



Instruction.py

```
1 def instructions(context_variables):
2     user_name = context_variables["user_name"]
3     return f"Help the user, {user_name}, do whatever they want."
4
5 agent = Agent(
6     instructions=instructions
7 )
8 response = client.run(
9     agent=agent,
10    messages=[{"role": "user", "content": "Hi!"}],
11    context_variables={"user_name": "John"}
12 )
13 print(response.messages[-1]["content"])
```

Instruction.py

```
1 Hi John, how can I assist you today?
```

Swarm

Calling Functions



Function.py

```
1 def greet(context_variables, language):
2     user_name = context_variables["user_name"]
3     greeting = "Hola" if language.lower() == "spanish" else "Hello"
4     print(f"{greeting}, {user_name}!")
5     return "Done"
6
7 agent = Agent(
8     functions=[print_hello]
9 )
10
11 client.run(
12     agent=agent,
13     messages=[{"role": "user", "content": "Usa greet() por favor."}],
14     context_variables={"user_name": "John"}
15 )
```

Output

```
1 Hola, John!
```

Swarm

Function Schema



function_schema.py

```
1 def greet(name, age: int, location: str = "New York"):
2     """Greets the user. Make sure to get their name and age before
3     calling.
4     Args:
5         name: Name of the user.
6         age: Age of the user.
7         location: Best place on earth.
8     """
9     print(f"Hello {name}, glad you are {age} in {location}!")
```

Output

```
1 {
2     "type": "function",
3     "function": {
4         "name": "greet",
5         "description": "Greets the user. Make sure to get their name and
6         age before calling.\n\nArgs:\n  name: Name of the user.\n  age: Age
7         of the user.\n  location: Best place on earth.",
8         "parameters": {
9             "type": "object",
10            "properties": {
11                "name": {"type": "string"},
12                "age": {"type": "integer"},
13                "location": {"type": "string"}
14            },
15            "required": ["name", "age"]
16        }
17    }
```

www.masteringllm.com



LLM Interview Course



Want to Prepare yourself for an LLM Interview?

- ✓ **120+** Questions spanning **14 categories** with Real Case Studies
- ✓ Curated **100+ assessments** for each category
- ✓ Well-researched **real-world interview questions based on FAANG & Fortune 500 companies**
- ✓ Focus on Visual learning
- ✓ Certification



Coupon Code - LLM50

Coupon is valid till 30th Oct 2024

AgenticRAG with LlamaIndex

Want to learn why AgenticRAG is future of RAG?

- ✓ Master **RAG fundamentals** through practical case studies
- ✓ Understand how to overcome **limitations of RAG**
- ✓ Introduction to **AgenticRAG** & techniques like **Routing Agents, Query planning agents, Structure planning agents, and React agents with human in loop.**
- ✓ **5 real-time case studies with code walkthroughs**