Quick Start Guide for uAgents Framework

Thisquickstart guide quickly let you install the uAgents Framework and get started in few steps!

Installation

System requirements

- Python 3.8+
- : A popular programming language.
- PIP
- : Python package manager for installing libraries.
- Poetry (optional)
- : For managing virtual environments and dependencies.

Installation steps

1. Create aProject Directory
2. . Open your terminal and create a new directory for your project:
3. mkdir directory_name
4. cd directory_name
5. Initialize and activate virtual environment. UsePoetry
6. to initialize and activate a virtual environment (optional but recommended):
7. poetry init
8. -
9. n && poetry shell
10. InstalluAgents Framework
11. . UsePIP
12. to install the uAgents Framework package.
13. pip install uagents
14. Verify installation. Check if the installation was successful.
15. pip show uagents

Troubleshooting

Problem on MacOS/Python 3.11 : Installing coincurve (17.0.0) fails.

Solution

Install the necessary tools.

brew install automake autoconf libtool

Usingon_message

andon_interval even handlers

Interval Task Example

This example shows how to make an agent perform a task periodically.

Create Script

Create a new Python script:

touch interval_task . py

Edit Script

Openinterval_task.py in your text editor and add the following code:

interval_task.py from uagents import Agent , Context

Create an agent named Alice

alice

Agent (name = "alice" , seed = "alice recovery phrase")

Define a periodic task for Alice

@alice . on_interval (period = 2.0) async

def

say_hello (ctx : Context): ctx . logger . info (f 'hello, my name is { ctx.name } ')

Run the agent

if

name

==

"**main**" : alice . run ()

Run Script

Run the script to see the output:

python interval

task . py Expected Output :hello, my name is alice every 2 seconds .

Message Handling Example

This guide will walk you through creating a simple interaction between two agents, Emma and Liam. Emma will send a message to Liam at regular intervals, and Liam will handle and log the received messages.

Create Script

Create a new Python script:

touch simple_example . py

Edit Script

Opensimple_example.py in your text editor and add the following code:

simple_example.py from uagents import Agent , Bureau , Context , Model

Define the message structure

class

Message (Model): message :

str

Create agents

emma

Agent (name = "Emma" , seed = "emma_seed") liam =

Agent (name = "Liam" , seed = "liam_seed")

Define behavior for Emma

```python
@emma . on_interval (period = 3.0 ) async
def
send_message ( ctx : Context):
```

Create an instance of the Message class with the desired content

message_to_liam

```python
Message (message = "Hey Liam, how's it going?" )
```

Send the message to Liam

```python
await ctx . send (liam.address, message_to_liam)
```

Define behavior for handling messages received by Liam

```python
@liam . on_message (model = Message) async
def
liam_message_handler ( ctx : Context ,
sender :
str ,
msg : Message): ctx . logger . info ( f "Received message from { sender } : { msg.message } " )
```

Create a bureau and add agents

bureau

```python
Bureau () bureau . add (emma) bureau . add (liam)
```

Run the bureau

```python
if
```

name

```python
==
```

"**main**" : bureau . run ()

Run Script

Run the script to see the agents communicate:

python simple_example . py Expected Output :

[Liam]: Received message from agent1q2aexr7d0e0x6z5r8zs972jyd8uq9429tc2qfgpax96x6ppyjs25keexnuk: Hey Liam, how's it going?

Reach out to the Team!

Cool! You are now ready to start exploring the concepts and resources available to start developing your agents on the Fetch Network! Remember that our Team is available on Telegram ↗(opens in a new tab) and Discord ↗(opens in a new tab) channels for any additional inquiry.

Was this page helpful?

Guides What's an Agent?