

# Creating your first agent

## Introduction

Once you've [installed](#) the uAgents library, it's quite simple to get a minimal use case running. Let our first agent be a simple inialisation and printing out the agent's name.

### The agent:

```
1. Let's create a Python script for this task, and name it by running:touch agent.py
2. We then need to import theAgent
3. andContext
4. classes from theuagents
5. library, and then create an agent using the classAgent
6. :
7. from
8. uagents
9. import
10. Agent
11. ,
12. Context
13. alice
14. =
15. Agent
16. (name
17. =
18. "alice"
19. , seed
20. =
21. "alice recovery phrase"
22. )
23. It is optional but useful to include aseed
24. parameter when creating an agent to set fixedaddresses
25. . Otherwise, random addresses will be generated every time you run the agent. Your address is kind of important, as
    this is how other agents will identify you.
26. Let's define asay_hello()
27. function for our agent to print a message periodically sayinghello, my name is ...
28. :
29. @alice
30. .
31. on_event
32. (
33. "startup"
34. )
35. async
36. def
37. say_hello
38. (
39. ctx
40. :
41. Context):
42. ctx
43. .
44. logger
45. .
46. info
47. (
48. f
49. 'hello, my name is
50. {
51. ctx.name
52. }
53. '
54. )
55. if
56. name
57. ==
```

```

58. "main"
59. :
60. alice
61. .
62. run
63. ()
64. The.on_event("startup")
65. decorator defines a behavior for this agent when it is run. In this case, the agent will execute thesay_hello()
66. function when the agent starts. TheContext
67. object is a collection of data and functions related to the agent. In this case, we just use the agent'sname
68. ,alice
69. . The agent executes the function and uses thectx.logger.info()
70. method to print the message.
71. Save the script.

```

The overall script should look as follows:

agent.py from uagents import Agent , Context

## alice

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

```
@alice . on_event ( "startup" ) async
```

```
def
```

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

```
if
```

```
name
```

```
==
```

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

## Run your agent

Make sure to have activated your virtual environment correctly.

Run the script:python agent.py

The output would be:

[alice]: hello, my name is alice Congratulations, you have just created your first Agent! We recommend learning about[interval tasks](#) next, then if you want to move really fast going to the[Communicating with other Agents](#) guide after.

## Was this page helpful?

[Installing the uAgents Framework](#) [Creating an interval task](#)