Using agents storage function

Introduction

In this guide, we want to illustrate how storage functions are called and how to use them. We want to create an agent which gets a value from the storage (starting from 0) every second. Then prints it, and puts the new value back into the storage but increased by 1 unit.

Walk-through

1. First of all, let's create a Python script, and name it by running:touch storage.py
2. .
3. Then, we need to open the script in the text editor of choice and import the necessary classes,Agent
4. andContext
5. , from theuagents
6. library.
7. Let's then create an agent namedalice
8. which logs a message every second using the.on_interval()
9. decorator, indicating the current count. Theon_interval()
10. function takes aContext
11. object as a parameter: theContext
12. object contains astorage
13. attribute, which is used to store and retrieve data between method calls:
14. storage.py
15. from
16. uagents
17. import
18. Agent
19. ,
20. Context
21. alice
22. =
23. Agent
24. (name
25. =
26. "alice"
27. , seed
28. =
29. "alice recovery phrase"
30.)
31. @alice
32. .
33. on_interval
34. (period
35. =
36. 1.0
37.)
38. async
39. def
40. on_interval
41. (
42. ctx
43. :
44. Context):
45. current_count
46. =
47. ctx
48. .
49. storage
50. .
51. get
52. (
53. "count"
54.)
55. or
56. 0

57. ctx
58. .
59. logger
60. .
61. info
62. (
63. f
64. "My count is:
65. {
66. current_count
67. }
68. "
69.)
70. ctx
71. .
72. storage
73. .
74. set
75. (
76. "count"
77. , current_count
78. +
79. 1
80.)
81. if
82. **name**
83. ==
84. "**main**"
85. :
86. alice
87. .
88. run
89. ()
90. Here, theon_interval()
91. function retrieves the current count from the storage attribute using thectx.storage.get()
92. method. It prints thecurrent_count
93. value, and then increments it by1
94. , and stores the updated count back to the storage attribute using thectx.storage.set()
95. method. The current count is then logged using thectx.logger.info()
96. method.
97. Save the script.

Run the script

On your terminal, make sure you activated the virtual environment.

Run the script:python storage.py

The output should look as follows:

[alice]: My count is: 1 [alice]: My count is: 2 [alice]: My count is: 3 ...

Was this page helpful?

Getting an agent addresses   Registering in the Almanac Contract