Agentverse Functions: register a coin toss agent as a Function

Introduction

In the Agentverse Functions: register your services on the Agentverse! ↗ guide, you have gotten familiar with the creation of an agent that can be registered as a Function and then be used in DeltaV ↗ .

i Checkout the Functions ↗ guide for additional information needed for Agent Functions and their registration on the Agentverse and Fetch network.

Create your coin toss agent!

For this navigate to the Agentverse: My Agents ↗(opens in a new tab) tab and click on the + New Agent button:

When the dialog is open, select the Toss a Coin use case:

A new agent will be created for you:

Run your coin toss agent!

After clicking on the row of your newly created agent, you should be able to see the source code of your coin toss agent in the editor view:

agent.py

Here we demonstrate how we can create a simple coin toss agent that is compatible with DeltaV.

After running this agent, it can be registered to DeltaV on Agentverse My Agents tab. For registration you will have to use the agent's address.

import random

third party modules used in this example

from pydantic import Field from ai_engine import UAgentResponse , UAgentResponseType

class

CoinToss (Model): choice :

str

=

Field (description = "The choice. Must be heads or tails.")

coin_toss_protocol

Protocol ("CoinToss")

@coin_toss_protocol . on_message (model = CoinToss, replies = {UAgentResponse}) async

def

toss_coin (ctx : Context ,

sender :

str ,

msg : CoinToss): random_number = random . randint (0 , 1) if random_number ==

0 : coin_tossed =

"heads" else : coin_tossed =

"tails" if coin_tossed == msg . choice : message =

"You won!" else : message =

"You lost!" await ctx . send (sender, UAgentResponse (message = message, type = UAgentResponseType.FINAL))

agent . include (coin_toss_protocol, publish_manifest = True) Now click on theRun button in the upper right corner of the editor so that you have your coin toss agent up and running!

Register your coin toss agent function!

Similar to the thisguide ↗ , let's navigate to theAgentverse: My Agents ↗(opens in a new tab)tab. Here, click on your agent to show the Agent Editor and then click on theDeploy tab to start registering your agent function.

After clicking the+ New Function button, you will need to provide all details required. Fill the form out as follows:

- Fucntion title
- : just the name of your Agent Function. In this example let's call itCoin toss function
- Description
- : Super important to be as detailed as you can, as reasoning engine looks at descriptions to understand what your Function does. In this example we can specify something like this:Coin toss Function. Takes the "heads" or "tails" input from the user and based on it decides if the user won or lost.
- Application
- ,Protocol
- ,Model
- andField descriptions
- will be automatically populated based on the source code ofyour coin toss agent ↗

Let's find our service on DeltaV

Now, head toDeltaV ↗(opens in a new tab) and sign in.

First, in theWhat function would you like to assemble? bar you can provide a predefined objective; let's typeToss a coin .

After being redirected to the chat screen, you will be asked to select an option. As your objective task (Toss a coin) specified on the previous screen contained words related to thedescription of your coin toss agent ↗, your Coin Toss Function is listed as an option.

Let's select it.

After selecting the task, you will be asked if you want to pickheads ortails . TheAI Engine ↗ behind DeltaV asks this question based on the description. The choice must be heads or tails.

Select whichever option you feel like. In this case let's pickheads .

Then, you can confirm or reject the context that the AI Engine is planning to execute. Let's confirm it!

After your function has been executed you can see theYou won! orYou lost! message.

With that,you have gotten a Coin Toss Function which can be discovered and contacted with DeltaV. Awesome!

Was this page helpful?

Register Agentverse Functions Field descriptions for DeltaV