Writing a Simple Service and Client (Python)

**Description:** This tutorial covers how to write a service and client node in python.  
  
**Tutorial Level:** BEGINNER  
  
**Next Tutorial:**  [Examining the simple service and client](http://wiki.ros.org/ROS/Tutorials/ExaminingServiceClient) 

catkin rosbuild

**Contents**

1. [Writing a Service Node](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.Writing_a_Service_Node)
   1. [The Code](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.The_Code)
   2. [The Code Explained](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.The_Code_Explained)
2. [Writing the Client Node](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.Writing_the_Client_Node)
   1. [The Code](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.The_Code-1)
   2. [The Code Explained](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.The_Code_Explained-1)
3. [Building your nodes](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.Building_your_nodes)
4. [Video Tutorial](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#Video_Tutorial)

Writing a Service Node

Here we'll create the service ("add\_two\_ints\_server") node which will receive two ints and return the sum.

Change directory into the beginner\_tutorials package, you created in the earlier tutorial, [creating a package](http://wiki.ros.org/ROS/Tutorials/CreatingPackage%3Fbuildsystem%3Dcatkin):

$ roscd beginner\_tutorials

Please make sure you have followed the directions in the previous tutorial for creating the service needed in this tutorial, [creating the AddTwoInts.srv](http://wiki.ros.org/ROS/Tutorials/CreatingMsgAndSrv#Creating_a_srv) (be sure to choose the right version of build tool you're using at the top of wiki page in the link).

The Code

Create the **scripts/add\_two\_ints\_server.py** file within the beginner\_tutorials package and paste the following inside it:

[Toggle line numbers](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29)

[1](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_1) #!/usr/bin/env python

[2](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_2)

[3](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_3) from beginner\_tutorials.srv import AddTwoInts,AddTwoIntsResponse

[4](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_4) import rospy

[5](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_5)

[6](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_6) def handle\_add\_two\_ints(req):

[7](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_7) print "Returning [%s + %s = %s]"%(req.a, req.b, (req.a + req.b))

[8](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_8) return AddTwoIntsResponse(req.a + req.b)

[9](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_9)

[10](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_10) def add\_two\_ints\_server():

[11](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_11) rospy.init\_node('add\_two\_ints\_server')

[12](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_12) s = rospy.Service('add\_two\_ints', AddTwoInts, handle\_add\_two\_ints)

[13](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_13) print "Ready to add two ints."

[14](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_14) rospy.spin()

[15](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_15)

[16](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_16) if \_\_name\_\_ == "\_\_main\_\_":

[17](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-d8bd520cf38bcf1cd93315e08496f17643f0723b_17) add\_two\_ints\_server()

Don't forget to make the node executable:

* chmod +x scripts/add\_two\_ints\_server.py

The Code Explained

Now, let's break the code down.

There's very little to writing a service using [rospy](http://wiki.ros.org/rospy). We declare our node using init\_node() and then declare our service:

[Toggle line numbers](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29)

[12](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-4af5a2319367916304ba4d0dd6f697e4c2f5788c_12) s = rospy.Service('add\_two\_ints', AddTwoInts, handle\_add\_two\_ints)

This declares a new service named add\_two\_ints with the AddTwoInts service type. All requests are passed to handle\_add\_two\_ints function. handle\_add\_two\_ints is called with instances of AddTwoIntsRequest and returns instances of AddTwoIntsResponse.

Just like with the subscriber example, rospy.spin() keeps your code from exiting until the service is shutdown.

Writing the Client Node

The Code

Create the **scripts/add\_two\_ints\_client.py** file within the beginner\_tutorials package and paste the following inside it:

[Toggle line numbers](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29)

[1](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_1) #!/usr/bin/env python

[2](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_2)

[3](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_3) import sys

[4](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_4) import rospy

[5](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_5) from beginner\_tutorials.srv import \*

[6](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_6)

[7](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_7) def add\_two\_ints\_client(x, y):

[8](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_8) rospy.wait\_for\_service('add\_two\_ints')

[9](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_9) try:

[10](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_10) add\_two\_ints = rospy.ServiceProxy('add\_two\_ints', AddTwoInts)

[11](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_11) resp1 = add\_two\_ints(x, y)

[12](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_12) return resp1.sum

[13](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_13) except rospy.ServiceException, e:

[14](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_14) print "Service call failed: %s"%e

[15](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_15)

[16](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_16) def usage():

[17](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_17) return "%s [x y]"%sys.argv[0]

[18](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_18)

[19](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_19) if \_\_name\_\_ == "\_\_main\_\_":

[20](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_20) if len(sys.argv) == 3:

[21](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_21) x = int(sys.argv[1])

[22](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_22) y = int(sys.argv[2])

[23](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_23) else:

[24](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_24) print usage()

[25](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_25) sys.exit(1)

[26](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_26) print "Requesting %s+%s"%(x, y)

[27](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-369fedb1315e4507f4b59d2e3b2a137ae63ab235_27) print "%s + %s = %s"%(x, y, add\_two\_ints\_client(x, y))

Don't forget to make the node executable:

$ chmod +x scripts/add\_two\_ints\_client.py

The Code Explained

Now, let's break the code down.

The client code for calling services is also simple. For clients you don't have to call init\_node(). We first call:

[Toggle line numbers](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29)

[8](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-9230d01f45af4649ee54ce639f7c09dc5ba16698_8) rospy.wait\_for\_service('add\_two\_ints')

This is a convenience method that blocks until the service named add\_two\_ints is available. Next we create a handle for calling the service:

[Toggle line numbers](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29)

[10](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-b0b6cdf636d5aadc585acc35117f3e34dd9c4276_10) add\_two\_ints = rospy.ServiceProxy('add\_two\_ints', AddTwoInts)

We can use this handle just like a normal function and call it:

[Toggle line numbers](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29)

[11](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-b52e2d1fa20a2403ab030cc359489fe0daeb1cfb_11) resp1 = add\_two\_ints(x, y)

[12](http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29#rospy_tutorials.2BAC8-Tutorials.2BAC8-WritingServiceClient.CA-b52e2d1fa20a2403ab030cc359489fe0daeb1cfb_12) return resp1.sum

Because we've declared the type of the service to be AddTwoInts, it does the work of generating the AddTwoIntsRequest object for you (you're free to pass in your own instead). The return value is an AddTwoIntsResponse object. If the call fails, a rospy.ServiceException may be thrown, so you should setup the appropriate try/except block.

Building your nodes

We use CMake as our build system and, yes, you have to use it even for Python nodes. This is to make sure that the [autogenerated Python code for messages and services](http://www.ros.org/wiki/ROS/Tutorials/CreatingMsgAndSrv#Creating_a_srv) is created.

Go to your catkin workspace and run catkin\_make.

# In your catkin workspace

$ cd ~/catkin\_ws

$ catkin\_make

Now that you have written a simple service and client, let's [examine the simple service and client](http://wiki.ros.org/ROS/Tutorials/ExaminingServiceClient).