

ROS2 Services Cheat Sheet

ROS2 Robotics Developer Course - Using ROS2 In Python



Create Service File

Example: srv/SrvName.srv

```
1 float32 my_request
2 ---
3 sensor_msgs/Image my_response
```



Modify package.xml

Include IDL Message Generation Dependencies

```
<build_depend>roscpp</build_depend>
<exec_depend>roscpp</exec_depend>
<member_of_group>roscpp</member_of_group>
```



Modify CMakeLists.txt

Include IDL Generation, Any Message Dependencies

```
find_package(roscpp REQUIRED)
find_package(sensor_msgs REQUIRED)
```

Include Custom Service Files, Dependencies

```
roscpp_generate_interfaces(${PROJECT_NAME}
  "srv/SrvName.srv"
  DEPENDENCIES sensor_msgs
)
```



Create Python Scripts

Import Custom Services

```
from my_pkg_name.srv import SrvName
```

Create Service Server Node

```
class MyServiceServer(Node):
    def __init__(self):
        super().__init__('my_service_server_node')
        self.srv = self.create_service(SrvName, "srv_topic_name",
                                       self.server_callback)

    def server_callback(self, request, response):
        res = do_something(request.my_request)
        response.my_response = res
        return response
```

Create Service Client Node

```
class MyServiceClient(Node):
    def __init__(self):
        super().__init__('my_service_client_node')
        self.client = self.create_client(SrvName, 'srv_topic_name')
        self.req = SrvName.Request()

    def send_request(self, my_user_input):
        self.req.my_request = my_user_input
        self.client.wait_for_service()

        self.future = self.client.call_async(self.req)
        rclpy.spin_until_future_complete(self, self.future)
        self.result = self.future.result()

        return self.result
```