

Creating a ROS Package

Created by Hind Aljuaid

Before starting, please complete [this tutorial](#)

Creating a catkin package

1. Type the following command then click enter.

```
cd /home/youruser/catkin_ws/src/
```

Hint: Don't forget to replace the word "youruser" in the previous command with youruser.

```
hind@hind-virtual-machine: ~/catkin_ws/src  
hind@hind-virtual-machine: ~$ cd /home/hind/catkin_ws/src/
```

2. After that, type the following command then click enter.

```
ls
```

```
hind@hind-virtual-machine: ~/catkin_ws/src$ ls  
CMakeLists.txt  
hind@hind-virtual-machine: ~/catkin_ws/src$
```

3. Then, type the following command then click enter.

```
catkin_create_pkg beginner_tutorials std_msgs rospy roscpp
```

```
hind@hind-virtual-machine: ~/catkin_ws/src$ catkin_create_pkg beginner_tutorials  
std_msgs rospy roscpp  
Created file beginner_tutorials/package.xml  
Created file beginner_tutorials/CMakeLists.txt  
Created folder beginner_tutorials/include/beginner_tutorials  
Created folder beginner_tutorials/src  
Successfully created files in /home/hind/catkin_ws/src/beginner_tutorials. Pleas  
e adjust the values in package.xml.
```

4. Now type the following command then click enter.

```
ls
```

```
hind@hind-virtual-machine: ~/catkin_ws/src$ ls  
beginner_tutorials CMakeLists.txt  
hind@hind-virtual-machine: ~/catkin_ws/src$
```

Building a catkin workspace and sourcing the setup file

Now open a new terminal.

1. Type the following command then click enter.

```
cd /home/youruser/catkin_ws/
```

Hint: Don't forget to replace the word "youruser" in the previous command with youruser.

```
hind@hind-virtual-machine: ~/catkin_ws
hind@hind-virtual-machine:~$ cd /home/hind/catkin_ws/
hind@hind-virtual-machine:~/catkin_ws$
```

2. Then, type the following command then click enter.

```
catkin_make
```

```
hind@hind-virtual-machine:~/catkin_ws$ catkin_make
Base path: /home/hind/catkin_ws
Source space: /home/hind/catkin_ws/src
Build space: /home/hind/catkin_ws/build
Devel space: /home/hind/catkin_ws/devel
Install space: /home/hind/catkin_ws/install
####
#### Running command: "cmake /home/hind/catkin_ws/src -DCATKIN_DEVEL_PREFIX=/home/hind/catkin_ws/devel -DCMAKE_INSTALL_PREFIX=/home/hind/catkin_ws/install -G Unix Makefiles" in "/home/hind/catkin_ws/build"
####
-- Using CATKIN_DEVEL_PREFIX: /home/hind/catkin_ws/devel
-- Using CMAKE_PREFIX_PATH: /opt/ros/kinetic
-- This workspace overlays: /opt/ros/kinetic
-- Found PythonInterp: /usr/bin/python2 (found suitable version "2.7.12", minimum required is "2")
-- Using PYTHON_EXECUTABLE: /usr/bin/python2
-- Using Debian Python package layout
-- Using empy: /usr/bin/empy
-- Using CATKIN_ENABLE_TESTING: ON
-- Call enable_testing()
-- Using CATKIN_TEST_RESULTS_DIR: /home/hind/catkin_ws/build/test_results
-- Found gtest sources under '/usr/src/gmock': gtests will be built
```

3. Next, type the following command then click enter.

```
. ~/catkin_ws/devel/setup.bash
```

```
hind@hind-virtual-machine:~/catkin_ws$ . ~/catkin_ws/devel/setup.bash
hind@hind-virtual-machine:~/catkin_ws$
```

Package dependencies and customizing the package

Now open a new terminal.

1. Type the following command then click enter.

```
cd /home/youruser/catkin_ws/src/beginner_tutorials/
```

Hint: Don't forget to replace the word "youruser" in the previous command with youruser.

```
hind@hind-virtual-machine: ~/catkin_ws/src/beginner_tutorials
hind@hind-virtual-machine:~$ cd /home/hind/catkin_ws/src/beginner_tutorials/
```

2. Then, type the following command then click enter.

```
ls
```

```
hind@hind-virtual-machine:~/catkin_ws/src/beginner_tutorials$ ls
CMakeLists.txt  include  package.xml  src
hind@hind-virtual-machine:~/catkin_ws/src/beginner_tutorials$
```

3. Next, type the following command then click enter.

```
cat package.xml
```

```
hind@hind-virtual-machine:~/catkin_ws/src/beginner_tutorials$ cat package.xml
<?xml version="1.0"?>
<package format="2">
  <name>beginner_tutorials</name>
  <version>0.0.0</version>
  <description>The beginner_tutorials package</description>

  <!-- One maintainer tag required, multiple allowed, one person per tag -->
  <!-- Example:  -->
  <!-- <maintainer email="jane.doe@example.com">Jane Doe</maintainer> -->
  <maintainer email="hind@todo.todo">hind</maintainer>

  <!-- One license tag required, multiple allowed, one license per tag -->
  <!-- Commonly used license strings: -->
  <!--   BSD, MIT, Boost Software License, GPLv2, GPLv3, LGPLv2.1, LGPLv3 -->
  <license>TODO</license>

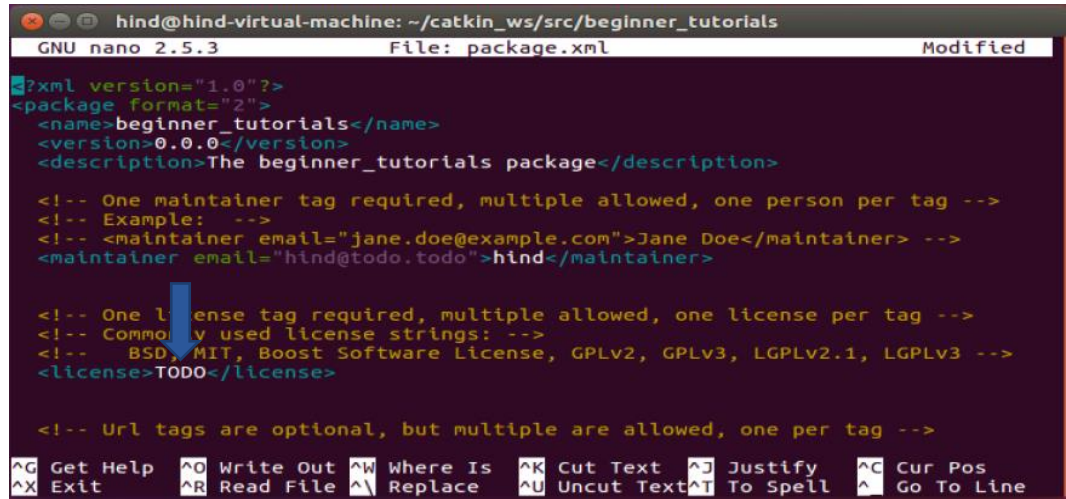
  <!-- Url tags are optional, but multiple are allowed, one per tag -->
  <!-- Optional attribute type can be: website, bugtracker, or repository -->
```

4. After that, type the following command then click enter.

```
nano package.xml
```

```
hind@hind-virtual-machine:~/catkin_ws/src/beginner_tutorials$ nano package.xml
```

- You will see it like this. Now replace "TODO" into "BSD". Then click ctrl-x



```
GNU nano 2.5.3 File: package.xml Modified
?xml version="1.0"?>
<package format="2">
  <name>beginner_tutorials</name>
  <version>0.0.0</version>
  <description>The beginner_tutorials package</description>

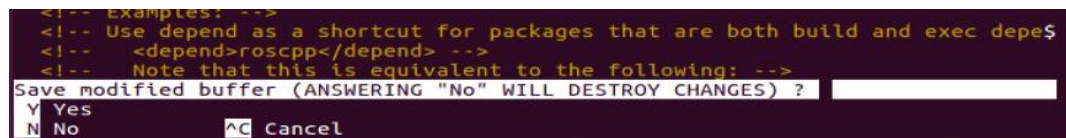
  <!-- One maintainer tag required, multiple allowed, one person per tag -->
  <!-- Example:  -->
  <!-- <maintainer email="jane.doe@example.com">Jane Doe</maintainer> -->
  <maintainer email="hind@todo.todo">hind</maintainer>

  <!-- One license tag required, multiple allowed, one license per tag -->
  <!-- Commonly used license strings: -->
  <!-- BSD, MIT, Boost Software License, GPLv2, GPLv3, LGPLv2.1, LGPLv3 -->
  <license>TODO</license>

  <!-- Url tags are optional, but multiple are allowed, one per tag -->

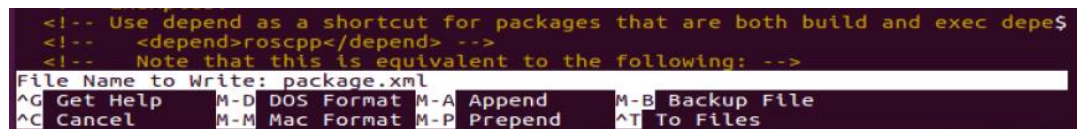
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line
```

- Now will ask you to save the modified, click y



```
<!-- Examples: -->
<!-- Use depend as a shortcut for packages that are both build and exec depe$
<!-- <depend>roscpp</depend> -->
<!-- Note that this is equivalent to the following: -->
Save modified buffer (ANSWERING "No" WILL DESTROY CHANGES) ?
Y Yes
N No ^C Cancel
```

- Now click ctrl-m



```
<!-- Use depend as a shortcut for packages that are both build and exec depe$
<!-- <depend>roscpp</depend> -->
<!-- Note that this is equivalent to the following: -->
File Name to Write: package.xml
^G Get Help ^M-D DOS Format ^M-A Append ^M-B Backup File
^C Cancel ^M-M Mac Format ^M-P Prepend ^M-T To Files
```

References

<http://wiki.ros.org/ROS/Tutorials/CreatingPackage>

<https://www.youtube.com/watch?v=RQXI6KVRlhM>