***Name: Shalini Das***

# Rostopics

1. rostopic echo
   echo <topic-name>
   Display messages published to a topic.
   `$ rostopic echo /topic_name`
2. rostopic list
   list
   Display a list of current topics.
   `$ rostopic list`
3. rostopic pub
   pub <topic-name> <topic-type> [data...]
   Publish data to a topic.
   `$ rostopic pub /topic_name std_msgs/String hello`

# Rosservices

1. rosmsg show
   show <message type>
   Display the fields in a ROS message type. You may omit the package name of the type, in which case rosmsg will search for matching types in all packages. Example:
   `$ rosmsg show std_msgs/String`
2. rosservice node
   node <service-name>
   Display the name of the node that provides a particular service.
   `$ rosservice node /service_name`
3. rosservice list
   list
   List all the services that are currently available.
   `$ rosservice list`

# Rosactions

1. client.wait_for_server(): To wait until the action server has started up and started listening for goals.
2. goal = actionlib_tutorials.msg.FibonacciGoal(order=20): To Create a goal to send to the action server.
3. client.send_goal(goal): To send the goal to the action server.

## Navigation Messages

**nav_msgs** defines the common messages used to interact with the navigation stack.

## Geometry Messages

**geometry_msgs** provides messages for common geometric primitives such as points, vectors, and poses. These primitives are designed to provide a common data type and facilitate interoperability throughout the system.

## Sensor Messages

**sensor_msgs** defines messages for commonly used sensors, including cameras and scanning laser rangefinders.