

envisonment is preset and your my bashow file is sourced.

To sun a node within a package

=> To change the name of mode

nos oum [Package-nane] [Oldnode-nane] _name: [New-node-Nan]

6 Understanding ROS Topic

* Using orat-graph

=> 9 rest-graph Creates a dynamic graph of What's going on in the suptem.

-> orat-graph is part of the oracle package.

910 Sown siat-graph sut-graph

STo sin siat-graph)

[onos command] -h

{To get details chord the}

Bus command)

Grostopic echo [topic_namo]

Sto display massage being passed

through see the topic

grostopic type [topic-name]

I gretums the massage type of and be to pic being bublished

* Using Gostopic pub

Lopic currently adventised.

grostopic pub [topic] [msg-type] [cngs]

-91 -> To publish Stredy Stream of Command

-1 => to only publish one message ham

grostopic hz [topm-namo]

[Reports the order at which data?

> sight-Plot displays a scorolling time plot of the data published. On topic-

light - I the devoid

nosommonat plot oxyl-pot

STO grun onat plots

In this QUI you can enter! the topic varichle to plot grossenvice list

Slists all the Services

onosservice type [service mane] Sometum type of service)

onosservice Coll Iservice] [ango] STO COUL OF Services

> eg => enosservice coll /clean { Clears the background of} { turtle sim-node

example Jerossenvice type /space | grosson show)

* 9105 param

=> Allows you to Store and manipulate data on ROS parameter Server.

=> grospanam uses YAML mankup language for Syntax.

grospanam get/

To show his the contents of the contents of the

of purply land t

919t-Console

to display output from modes.

grate logger level

(DEBUG, WARN, INFO and ERROR) of nodes as they own

grosson sigt_Console sigt-Console)

enos orum orat-logger-level orat-logger-level

=> Logging level are porionitized in the following

Fatel -> Enggon -> Warm -> Info -> Oebug

By setting the logger level, you will get all messages of the poissity level on higher.

* Using moslaunch

910slaunch [Package_name] [filename.launch]

{ To Stant a launch file}

grosed Package nano filenano

STo edit the file in the parkages

=> Default editor of orosad is Vim.

If it is not then them,

export EDITOR = 'Mamo -W'

(1) Greating a ROS message (#2) (55)

*Msg > msg files are simple text files that describe the fields of a ROS mossage.

-> They are used to generale Source code for messages in different language.

* Sow > Am . Sow file describes a Service. It is Composed of two purts: a proquest & a proponse.

msg L> messege_nane.msg.

=> edit Package. xml k CMakeLists. +++ accordingly-

Sono La Service-man. Sono

=> edit Package.xml k CMukeLists.txt occardingly.

The street of the street of the

bringly and in medical in a practical to

(1) Woiting a Simple Publisher and Subscriber (C++)

* Woriting the Publisher Node

include " mos/mos!" Toosh is in no parkay.

#include " onos/onos.h"

include "std_msgs/Storing.h" Storing.h in Std_msgo

Hindlude (Sstoream)

Storing Stream

) > used to convict first into storing and istoring into inti

Storing storan SSO: 550>

int a = 55!

Stoles to: 1/9ntigu Stand in Stoles Stran Sso

SSO >> b; // String Stored in b from SSO-

⇒ A storing stream associates a storing object with a stream dlowing you to read from the storing as if it were a stream (11/2 Cin)

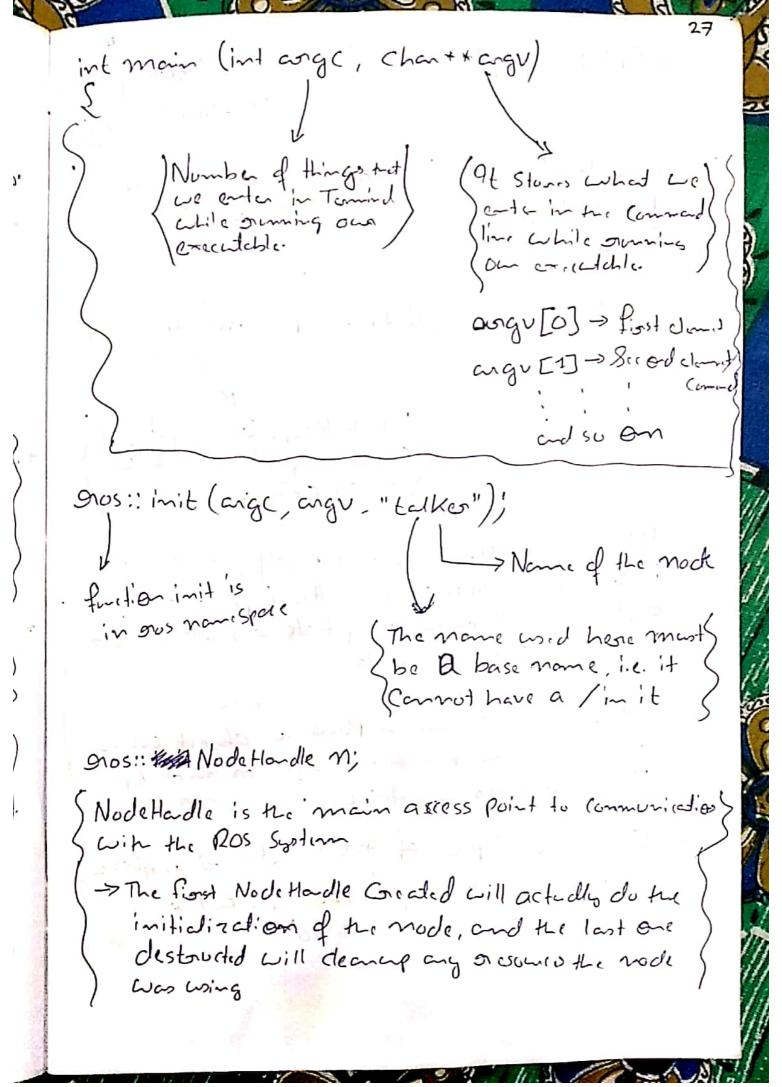
Basic methods are:

O Clean () => to clean the stroom

6 Stor() => to get and Set Stoins object whose content is present in storem.

3 K = add a staing to the staining stream object.

65 >> => gread Someting from the String Stream



ovs: Publisher chatter-Pub = M. adventise Ksid-msgs::Strig (Chatter", 1000)

- Posted you want to publish on a given topiger name.
- This invokes a coll to the Ros master node.

Chich Keeps a oregistors of) Culu is publishing & culo is subscribing

- is trying to subscribe to this topic mane, and thing will in turn negotiate a personto-person conscition with this rode.
- advertise () oreturns a Publishen object which does you to publish missage on that topic through a collecto publish ().
- Direct and destroyed, the lopic will be automatically unadventised.

=> Std_msyx: Storing is msg type => Chatter is name of topic => 1000 => Size of buff on

gnos::Rate Loup-grate (10).

-) => Rale object allows you to specify a frequery \
 that you would like to loop at,
 - Since the last coll to Rade: Sleep (), and Sleep for the Cornect amount of time.

pint Count=0; While (ones:: OK())

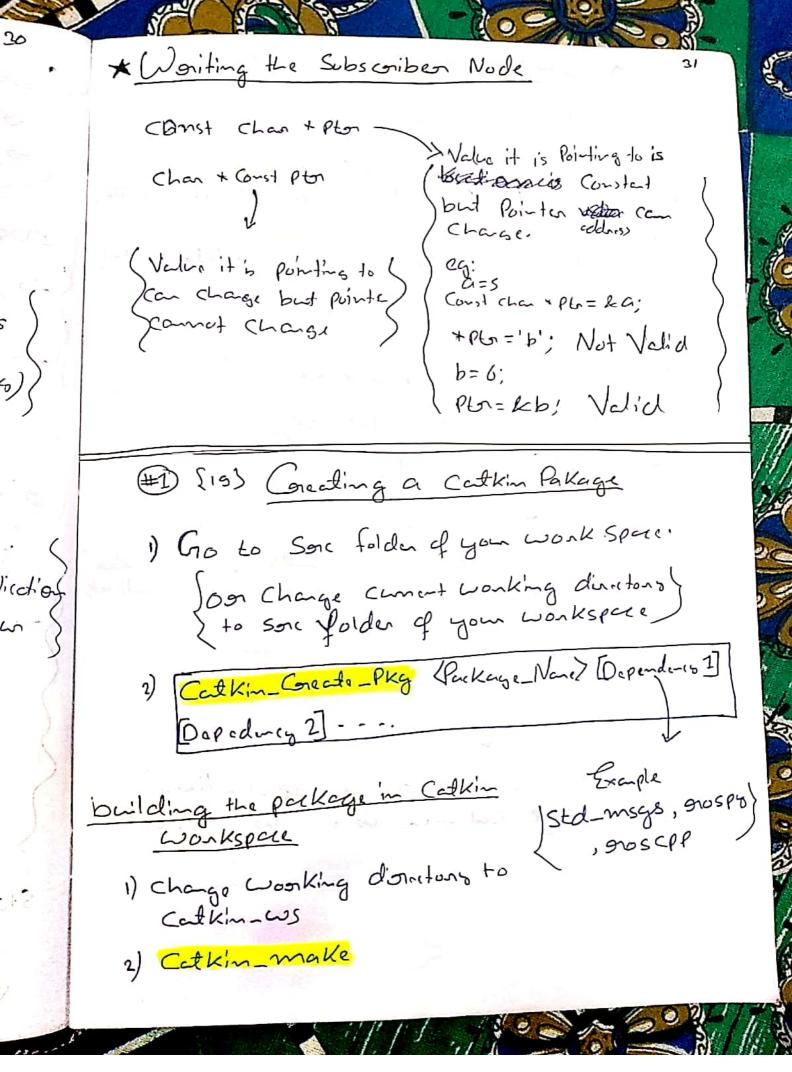
=> 9005:: OK (0) always ordums tome except if !-

- (1) Ctol + (is pressed
- by another node with the Same mane.
- (3) 9005: Sheddown() has been colled by another part of the application.
 (4) all 5105: Node Hadles has been distroyed.

Std-msgs: String msg

Smessage of name msg end type string in (Std-msgs name space

```
Std: Storgstram ss;
  SS << "hello world" << Count;
 msg.data= ss.str():
 ROS_INFO ("% S", msg. data: C_Stor());
> ROS_INFO is
                       (C-Stor oreturns a
  suplament for
                        (Const Chan * that points
   Bi-tf.
                        )to a null termineded
                        (Stoing (i.e. (skyle stoing)
 Chatton-Pub. Publish (msg);
 mos: Spinonce();
 => Calling Jus: Spinonce() hege is not necessary.
 => If we were to add a Subscription into this application
    , and did not have sos: Spin Once (), here your-
    Collbacks would never got alded.
loop-orde. Sleep ();
++ Count;
enctum 0;
          eter one i had
```



#include "snos/snos.h"

include "std-msgs! Stoing.h"

Void chettencallberk (const std-msgs:: Stoing:: ConstPtn/e msg)

ROSINFO ("I head Exs]", msg > deda. c_stol):

add_exe

int main (int aig (, chan +*ang V)

snos:: init (ang (, ang v, "listener");

snos:: Node Hondle handle;

Aft

and

gnos: Subscriben Sub;

Sub = handle. Subscribe ("chatter", 1000, chatter)

gnos: Spm()

) enters a loop, calling message callbacks as fast as possible.

> Building your nodes

Add the Polling in the CMakeLists. Ext

L msg)

add_executede (talker sonc/talker.cpp)
tanget_link_liboraries (talker \${Catkin_LIBRARIES)}
add_dependencies (talker test_pakage1-generat
-message-cpp)

add-executable (listenen son/listenen.cpp)

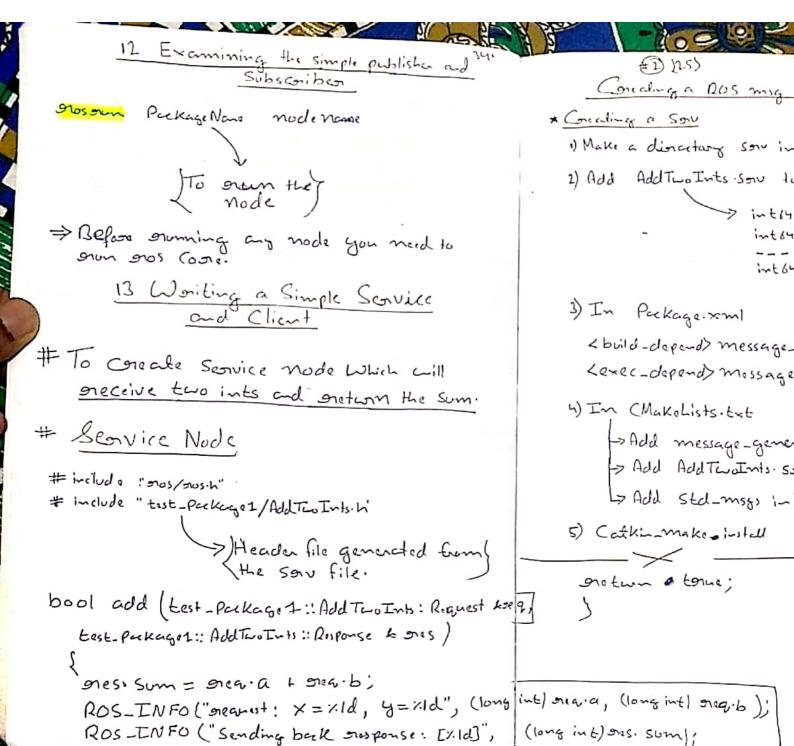
tanget-link-liboranies (listenen \${catkin_LIBRARIES)}

add-dependencies (listenen tast-parkage-generale-mossage-cop)

=> After that change pud to Gom Cathin Workspore and execute Cathin-make Command.

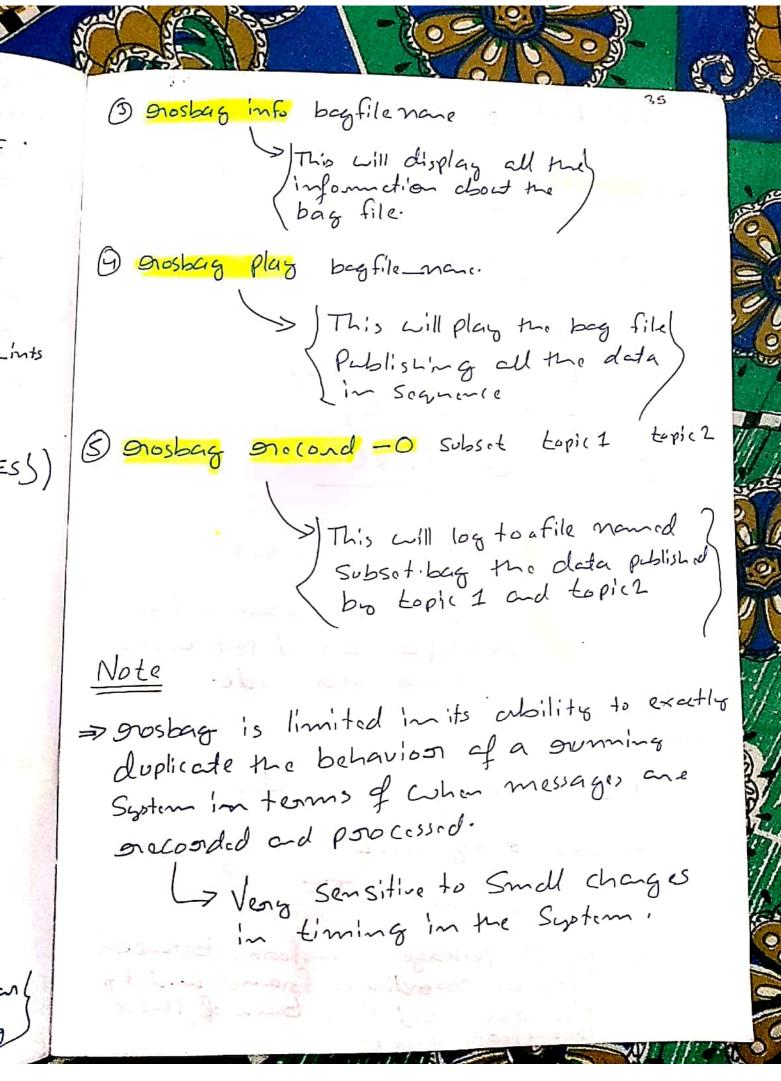
一一一大日 一一一一一一一一

thought in the late of the same is



int main (int ange, chan ** angu) 9009 900 Cli gros: init (angc, angv, "add-two-int-seaver"); enos: NodeHandle M Ea 0105: Service Server Service; Service = n.advertiseService ("add-two-ints", add): Sø ROS_INFO ("Ready to add two Ints"); 905: Spin (); onetwan 0; 0 Service name) # Waiting the Client Node #include "onos/oros." "test-Package 1/Add Tao Ints. h" # include # include LCS+dlide> >>) Foor Conventing Stains to inleger Int main (int ange, chan * + angu) onos: ! Init (ange, ange, "add-two-into-client"); if (angc!=3) ROS_INFO ("Usage: add-two_ints_client XY"); gretum 1;

gnos: NodeHandle M; gnos: Service Client client client = M. Service Client < test-package 1: Add Two Ints> ieaucs"): ("add-two-ints"); to long long int) testpakage1: AddTwInds Soni es", cod) Sow. orequest. a = atoll (argutis): Sov. Degrot.b = atoll (angu [2]): if (client. Cell (Sov)) This cetually cells the Service with argument sorv) ROS_INFO (" Sum: Y.Id", (long int) Sov. Disporse. Sum) else ROS_ERROR ("Failed to cell Service add_two_ints'); Jichum 1; gratum 0; ');

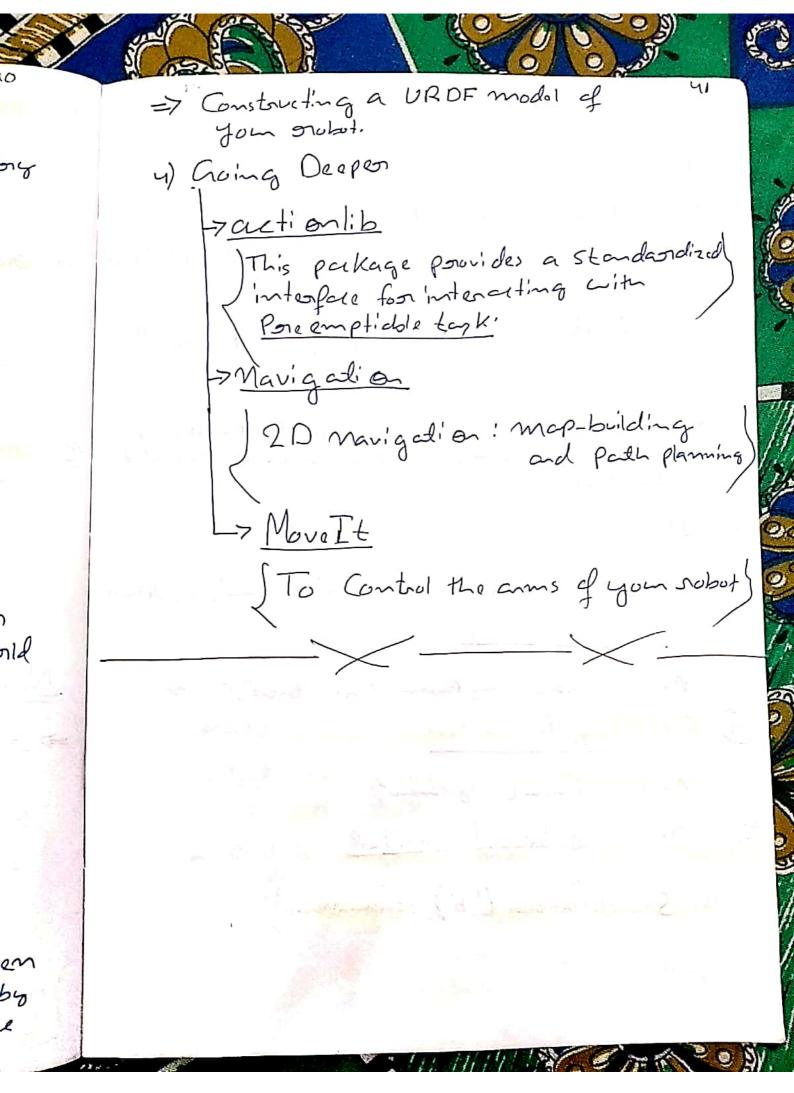


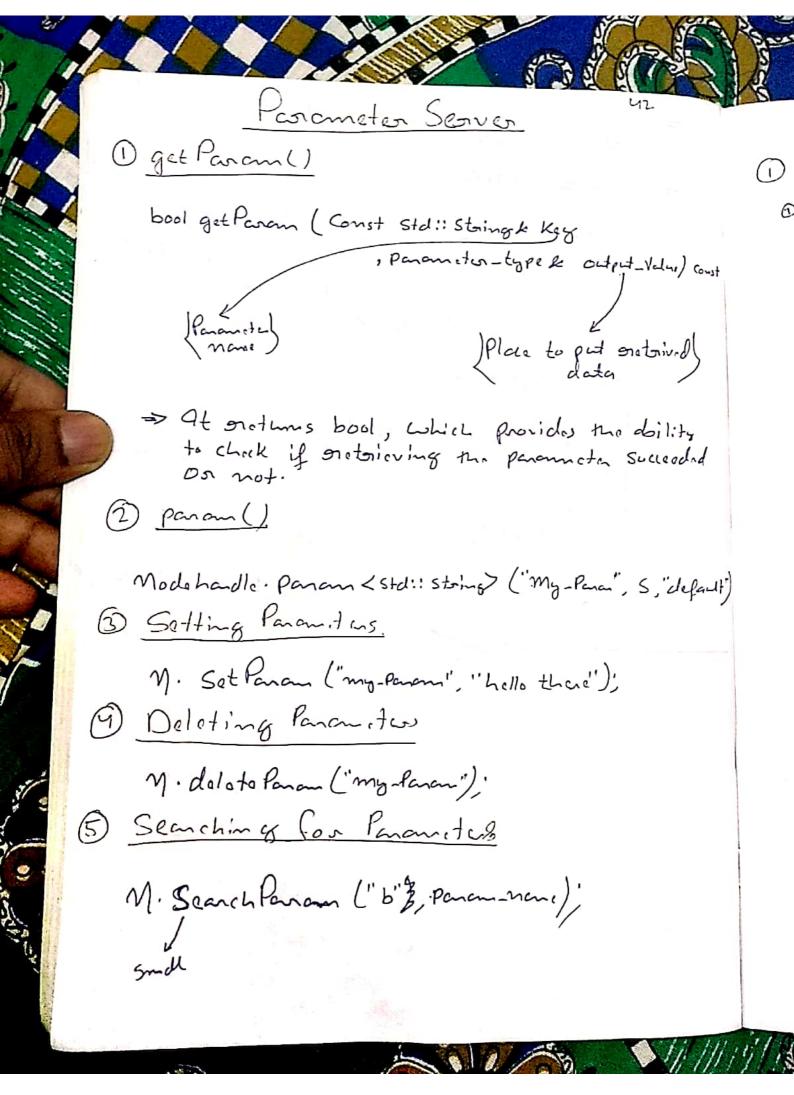
- (B) Getting Started wit soweth
- => groswelf examines your system to try and find problems.
 - 13) Navigating the ROS wiki > ROS Wiki Landing Page > ROS Packago Pages > ROS Stack Pages
 - 20 Where Next?
 - ROS) DRobot middle-Ware

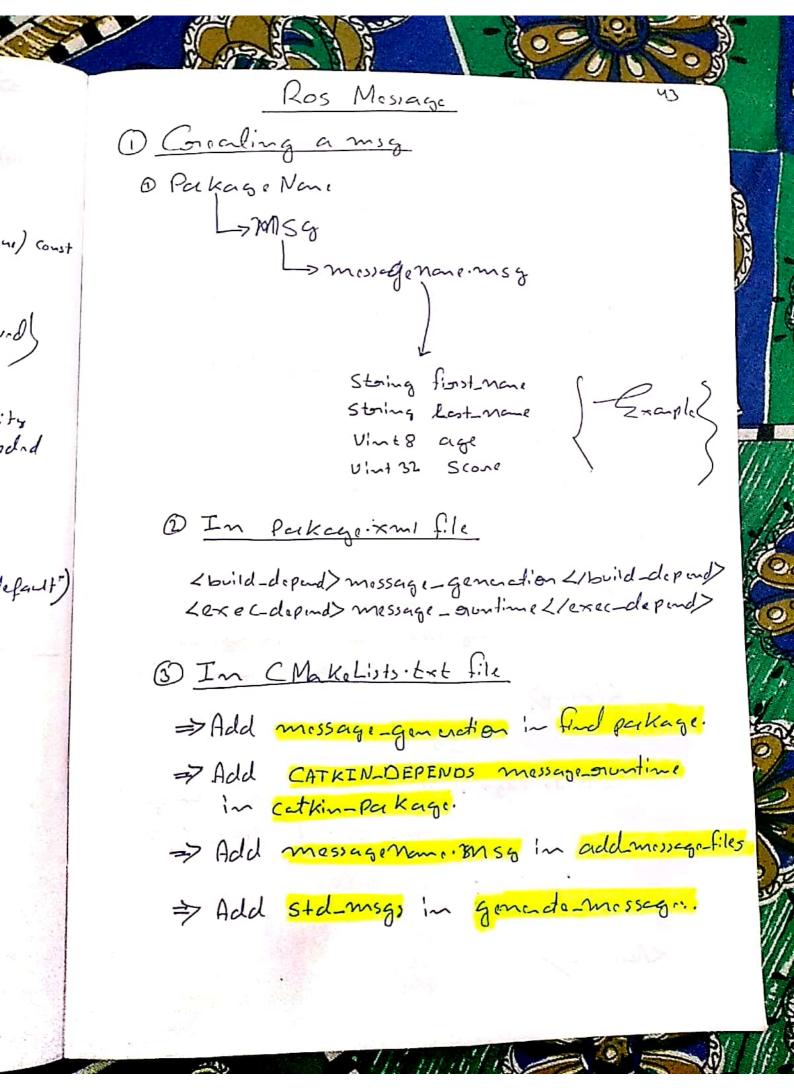
To main to share their code.

To main season to use ROS

- 1) Launching a Simulatoon
- 2) Explosing RViz
- 3) Understanding TF
- => The TF package to ansforms between different coordinate frames wood by your probot and Keops track of these to ansforms over time.







Laurch file 1 Using Groslaunch At Starts nodes as defined in a launch file gros laurch [Package] [filename.laurch] => launch files are Kept in launch directors Insido the pakage. 2 Laurel file. < 1 aun (h> (group ns="nan.spar"> Lnode Pkg="Pakagona." nan = 'nodonar" Eypa = "oniginal nodona" </rap> 09 Lnode Pkg = " Peckage neur" nan = "node neur give" Eypo = "Osigin Inudena." </launch>

