

⚠ Importante

Al haber realizado la instalación anterior de `Conda` en la *PC global mente*.

⚡ ¿Comó sabemos que si esta instalado o en uso?

- Si cerramos la terminal y abrimos una nuevamente, al inicio, justamente en la parte izquierda del nombre de usuario, veremos lo siguiente `(base)` .

```
1: iudc@iudc: ~ ~  
(base) iudc@iudc:~$ |
```

- El `(base)` , significa que si esta activado.

Ahora es importante saber, ya que en los pasos anteriores creamos entornos con `Conda` , si por algunas razones olvidaste como revisar que entorno creaste para activarlo. Se puede realizar de la siguiente manera:

```
conda env list  
# Ten en cuenta que el entorno base debe estar activado para ello.
```

- Nos da la siguiente salida en la terminal

```
1: iudc@iudc: ~ ~  
(base) iudc@iudc:~$ conda env list  
  
# conda environments:  
#  
# * -> active  
# + -> frozen  
base * /home/iudc/miniconda3  
amo /home/iudc/miniconda3/envs/amo  
env_isaaclab /home/iudc/miniconda3/envs/env_isaaclab  
g1_env /home/iudc/miniconda3/envs/g1_env  
g1_rl /home/iudc/miniconda3/envs/g1_rl  
isaaclab /home/iudc/miniconda3/envs/isaaclab  
isaaclab_legacy /home/iudc/miniconda3/envs/isaaclab_legacy  
locomani /home/iudc/miniconda3/envs/locomani  
mujoco_env /home/iudc/miniconda3/envs/mujoco_env  
unitree /home/iudc/miniconda3/envs/unitree  
unitree_lerobot /home/iudc/miniconda3/envs/unitree_lerobot  
unitree_py11 /home/iudc/miniconda3/envs/unitree_py11  
  
(base) iudc@iudc:~$ |
```

- Activamos el entorno:

```
conda activate isaaclab
```

```
1: iudc@iudc: ~ 
(base) iudc@iudc:~$ conda activate isaaclab
(isaaclab) iudc@iudc:~$ |
```

- Creamos la carpeta, si no lo hiciste vuelve al inicio de la documentación.

```
mkdir -p laboratorio_isaaclab
```

```
1: iudc@iudc: ~ 
(isaaclab) iudc@iudc:~$ mkdir -p laboratorio_isaaclab
(isaaclab) iudc@iudc:~$ |
```

- Ingresamos a la carpeta creada

```
cd laboratorio_isaaclab
```

```
1: iudc@iudc: ~/laboratorio_isaaclab ~ 
(isaaclab) iudc@iudc:~$ cd laboratorio_isaaclab/
(isaaclab) iudc@iudc:~/laboratorio_isaaclab$ |
```

- Ingresamos al entorno clonado:

```
cd IsaacLab/
```

```
1: iudc@iudc: ~/laboratorio_isaaclab/IsaacLab ~ 
(isaaclab) iudc@iudc:~/laboratorio_isaaclab$ cd IsaacLab/
(isaaclab) iudc@iudc:~/laboratorio_isaaclab/IsaacLab$ |
```

- Visualizar los entornos que podemos entrenar

```
./isaaclab.sh -p scripts/environments/list_envs.py
```

```
iudc@iudc:~/laboratorio_isaaclab/IsaacLab ~
| isaaclab_tasks.manager_based.manipulation.reach.config.ur_10.joint_pos_env_cfg:UR10ReachEnvCfg |
| 133 | Isaac-Reach-UR10-Play-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.reach.config.ur_10.joint_pos_env_cfg:UR10ReachEnvCfg_PLAY |
| 134 | Isaac-Stack-Cube-Franka-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.franka.stack_joint_pos_env_cfg:FrankaCubeStackEnvCfg |
| 135 | Isaac-Stack-Cube-Instance-Randomize-Franka-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.franka.stack_joint_pos_instance_randomize_env_cfg:FrankaCubeStackInstanceRandomizeEnvCfg |
| 136 | Isaac-Stack-Cube-Franka-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.franka.stack_ik_rel_env_cfg:FrankaCubeStackEnvCfg |
| 137 | Isaac-Stack-Cube-Franka-IK-Rel-Visuomotor-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.franka.stack_ik_rel_visuomotor_env_cfg:FrankaCubeStackVisuomotorEnvCfg |
| 138 | Isaac-Stack-Cube-Franka-IK-Rel-Visuomotor-Cosmos-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.franka.stack_ik_rel_visuomotor_cosmos_env_cfg:FrankaCubeStackVisuomotorCosmosEnvCfg |
| 139 | Isaac-Stack-Cube-Franka-IK-Abs-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.franka.stack_ik_abs_env_cfg:FrankaCubeStackEnvCfg |
| 140 | Isaac-Stack-Cube-Instance-Randomize-Franka-IK-Rel-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.franka.stack_ik_rel_instance_randomize_env_cfg:FrankaCubeStackInstanceRandomizeEnvCfg |
| 141 | Isaac-Stack-Cube-Franka-IK-Rel-Blueprint-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.franka.stack_ik_rel_blueprint_env_cfg:FrankaCubeStackBlueprintEnvCfg |
| 142 | Isaac-Stack-Cube-Franka-IK-Rel-Skillgen-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.franka.stack_ik_rel_env_cfg:SkillgenEnvCfg |
| 143 | Isaac-Stack-Cube-Bin-Franka-IK-Rel-Mimic-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.franka.bin_stack_ik_rel_env_cfg:FrankaBinStackEnvCfg |
| 144 | Isaac-Stack-Cube-Galbot-Left-Arm-Gripper-RmpFlow-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.galbot.stack_rmp_rel_env_cfg:RmpFlowGalbotLeftArmCubeStackEnvCfg |
| 145 | Isaac-Stack-Cube-Galbot-Right-Arm-Suction-RmpFlow-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.galbot.stack_rmp_rel_env_cfg:RmpFlowGalbotRightArmCubeStackEnvCfg |
| 146 | Isaac-Stack-Cube-Galbot-Left-Arm-Gripper-Visuomotor-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.galbot.stack_rmp_rel_env_cfg:RmpFlowGalbotLeftArmCubeStackVisuomotorEnvCfg |
| 147 | Isaac-Stack-Cube-Galbot-Left-Arm-Gripper-Visuomotor-Joint-Position-Play-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.galbot.stack_rmp_rel_env_cfg:GalbotLeftArmJointPositionCubeStackVisuomotorEnvCfg_PLAY |
| 148 | Isaac-Stack-Cube-Galbot-Left-Arm-Gripper-Visuomotor-RmpFlow-Play-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.galbot.stack_rmp_rel_env_cfg:GalbotLeftArmRmpFlowCubeStackVisuomotorEnvCfg_PLAY |
| 149 | Isaac-Stack-Cube-UR10-Long-Suction-IK-Rel-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.ur10_gripper.stack_ik_rel_env_cfg:UR10LongSuctionCubeStackEnvCfg |
| 150 | Isaac-Stack-Cube-UR10-Short-Suction-IK-Rel-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.manipulation.stack.config.ur10_gripper.stack_ik_rel_env_cfg:UR10ShortSuctionCubeStackEnvCfg |
| 151 | Isaac-Navigation-Flat-Anymal-C-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.navigation.config.anymal_c.navigation_env_cfg:NavigationEnvCfg |
| 152 | Isaac-Navigation-Flat-Anymal-C-Play-v0 | isaaclab.envs:ManagerBasedRLEnv |
| isaaclab_tasks.manager_based.navigation.config.anymal_c.navigation_env_cfg:NavigationEnvCfg_PLAY |
+-----+
(isaaclab) iudc@iudc:~/laboratorio_isaaclab/IsaacLab$ |
```

- Ahora vamos a ejecutar un entrenamiento

Terreno plano

- Va hasta 1500 iteraciones y puede que tarde 30 minutos a una hora

```
./isaaclab.sh -p scripts/reinforcement_learning/rsl_rl/train.py --task
Isaac-Velocity-Flat-G1-v0 --headless
```

```
#####
# Learning iteration 1/1500

    Computation: 144353 steps/s (collection: 0.644s, learning 0.037s)
    Mean action noise std: 1.00
    Mean value_function loss: 3.6480
    Mean surrogate loss: -0.0029
    Mean entropy loss: 52.5727
    Mean reward: -6.27
    Mean episode length: 47.02
Episode_Reward/track_lin_vel_xy_exp: 0.0063
Episode_Reward/track_ang_vel_z_exp: 0.0023
    Episode_Reward/lin_vel_z_l2: -0.0022
    Episode_Reward/ang_vel_xy_l2: -0.0543
    Episode_Reward/dof_torques_l2: -0.0022
    Episode_Reward/dof_acc_l2: -0.0049
    Episode_Reward/action_rate_l2: -0.0138
    Episode_Reward/feet_air_time: 0.0002
Episode_Reward/flat_orientation_l2: -0.0129
    Episode_Reward/dof_pos_limits: -0.0003
Episode_Reward/termination_penalty: -0.1956
    Episode_Reward/feet_slide: -0.0040
Episode_Reward/joint_deviation_hip: -0.0026
Episode_Reward/joint_deviation_arms: -0.0029
Episode_Reward/joint_deviation_fingers: -0.0018
Episode_Reward/joint_deviation_torso: -0.0008
Metrics/base_velocity/error_vel_xy: 0.0859
Metrics/base_velocity/error_vel_yaw: 0.4277
    Episode_Termination/time_out: 0.0270
    Episode_Termination/base_contact: 0.4604
-----
    Total timesteps: 196608
    Iteration time: 0.68s
    Time elapsed: 00:00:02
    ETA: 00:29:23
```

Terreno random

- Va hasta 3000 iteraciones y puede que tarde 1 hora

```
./isaaclab.sh -p scripts/reinforcement_learning/rsl_rl/train.py --task
Isaac-Velocity-Rough-G1-v0 --headless
```

```
#####
# Learning iteration 1/3000

          Computation: 118772 steps/s (collection: 0.770s, learning 0.058s)
          Mean action noise std: 1.00
          Mean value_function loss: 3.8041
          Mean surrogate loss: 0.0012
          Mean entropy loss: 52.5633
          Mean reward: -6.08
          Mean episode length: 47.31
Episode_Reward/track_lin_vel_xy_exp: 0.0069
Episode_Reward/track_ang_vel_z_exp: 0.0046
    Episode_Reward/lin_vel_z_l2: 0.0000
    Episode_Reward/ang_vel_xy_l2: -0.0518
    Episode_Reward/dof_torques_l2: -0.0002
    Episode_Reward/dof_acc_l2: -0.0064
    Episode_Reward/action_rate_l2: -0.0136
    Episode_Reward/feet_air_time: 0.0001
Episode_Reward/flat_orientation_l2: -0.0126
    Episode_Reward/dof_pos_limits: -0.0003
Episode_Reward/termination_penalty: -0.1944
    Episode_Reward/feet_slide: -0.0040
Episode_Reward/joint_deviation_hip: -0.0026
Episode_Reward/joint_deviation_arms: -0.0028
Episode_Reward/joint_deviation_fingers: -0.0018
Episode_Reward/joint_deviation_torso: -0.0008
    Curriculum/terrain_levels: 3.1227
Metrics/base_velocity/error_vel_xy: 0.0835
Metrics/base_velocity/error_vel_yaw: 0.4252
    Episode_Termination/time_out: 0.0266
    Episode_Termination/base_contact: 0.4166
-----
          Total timesteps: 196608
          Iteration time: 0.83s
          Time elapsed: 00:00:02
          ETA: 01:07:49
```

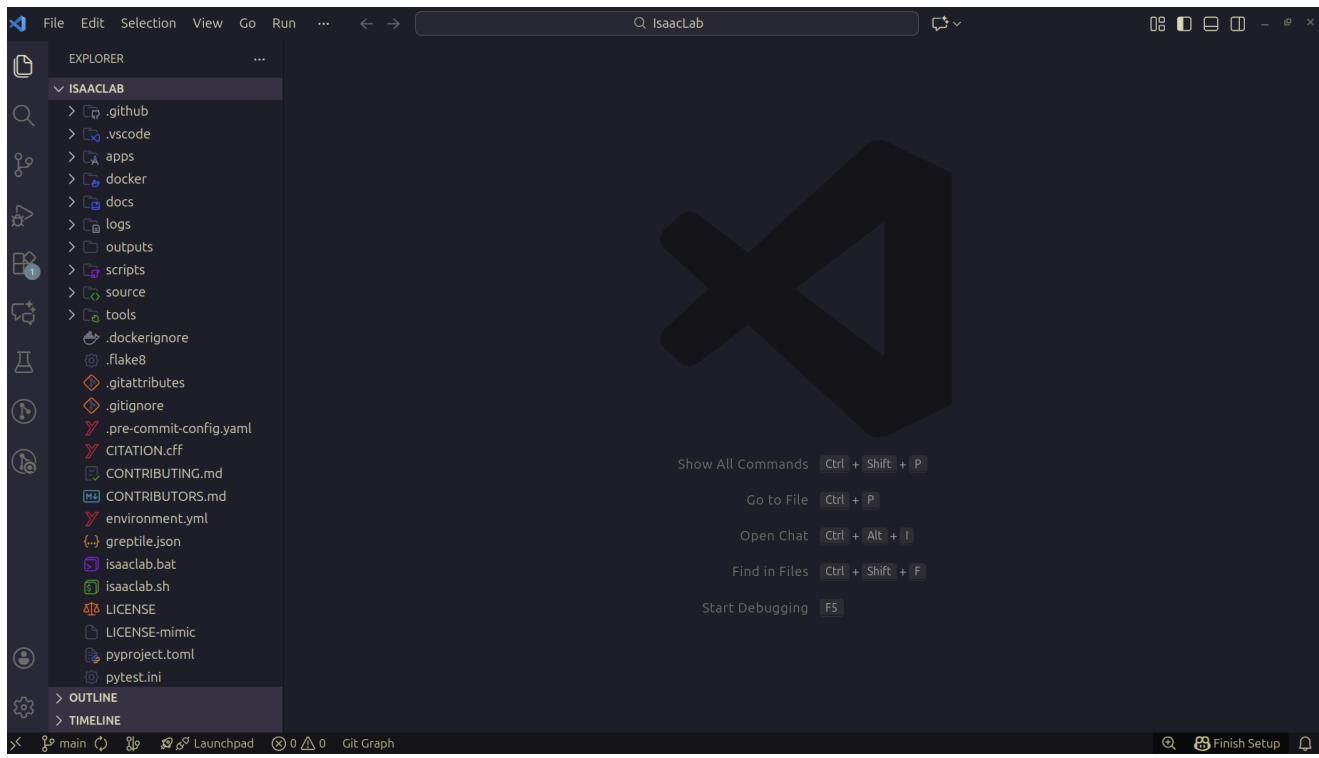
Ahora al entrenar nos genera unas carpetas junto con los puntos de guardado, donde se generara la politica a exportar:

- Nos mantenemos en la siguiente ubicacion `~/laboratorio_isaaclab` junto al entorno (`isaaclab`)

```
2:iudc@iudc: ~/laboratorio_isaaclab/IsaacLab ~
(isaaclab) iudc@iudc:~/laboratorio_isaaclab/IsaacLab$ |
```

- Importante tener instalado `VSCode`, en la ubicacion anterior de la terminal ingresamos con:

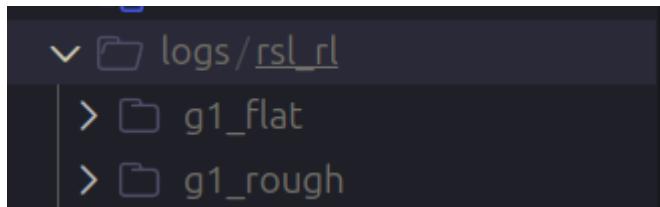
```
code .
```



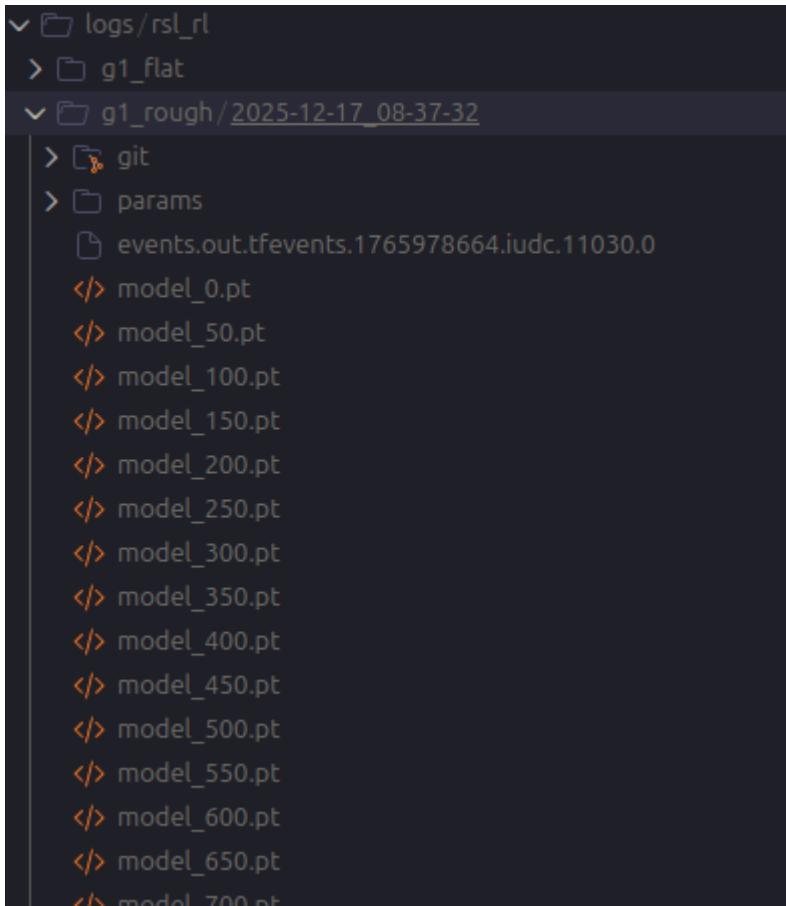
- Ahora la carpeta es logs



- Podemos ver que contiene carpetas, una con terreno plano y otro con random, son dos ya que se usaron de ejemplo para la documentación



- En las carpetas podremos ver los archivos con extension .pt osea pytorch



- Para visualizar un punto en específico usamos el siguiente comando, pero hay que tener en cuenta la ubicación/ruta del archivo

Ejemplo

- Ruta relativa:

```
logs/rsl_rl/g1_rough/2025-12-17_08-37-32/model_600.pt
```

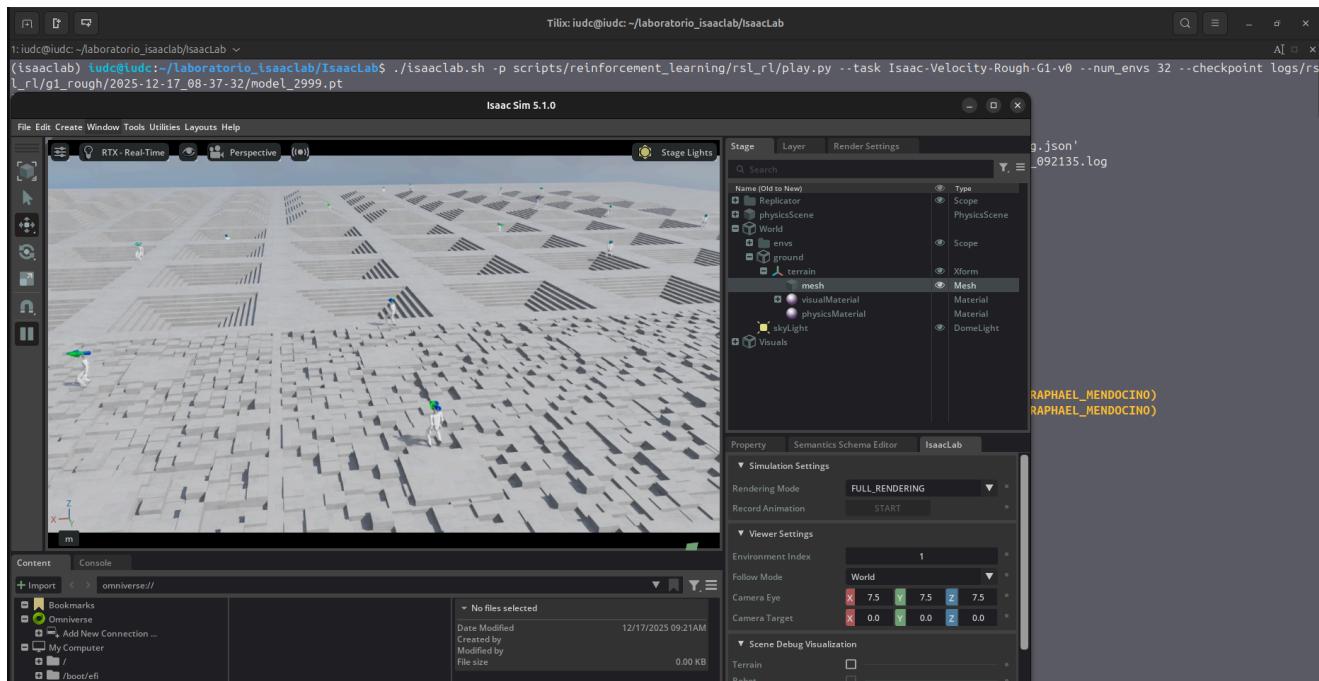
- Ruta absoluta:

```
/home/iudc/laboratorio_isaaclab/IsaacLab/logs/rsl_rl/g1_rough/2025-12-17_08-37-32/model_600.pt
```

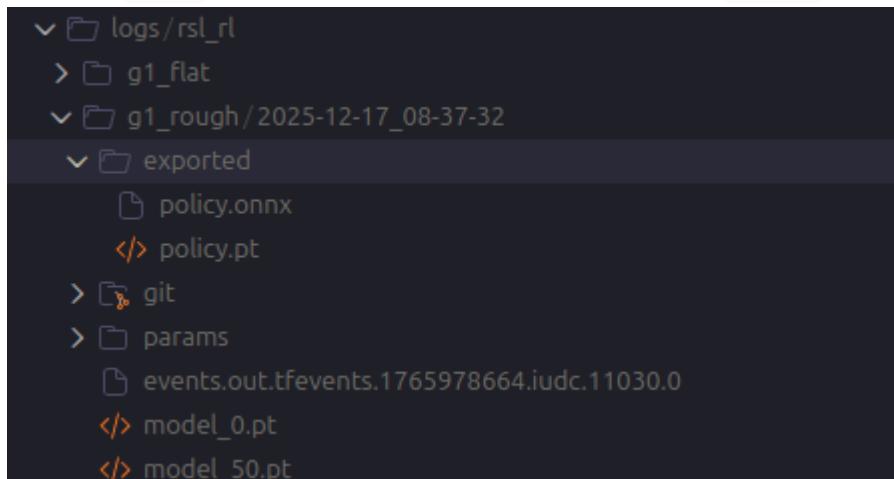
Con cualquiera funcionara solo es saber que entorno lanza si Rough o Flat para el lanzamiento del comando.

- Vamos a probar la ruta relativa para lanzar el punto guardado, por el entrenamiento del entorno Rough .

```
./isaaclab.sh -p scripts/reinforcement_learning/rsl_rl/play.py --task Isaac-Velocity-Rough-G1-v0 --num_envs 32 --checkpoint logs/rsl_rl/g1_rough/2025-12-17_08-37-32/model_2999.pt
```

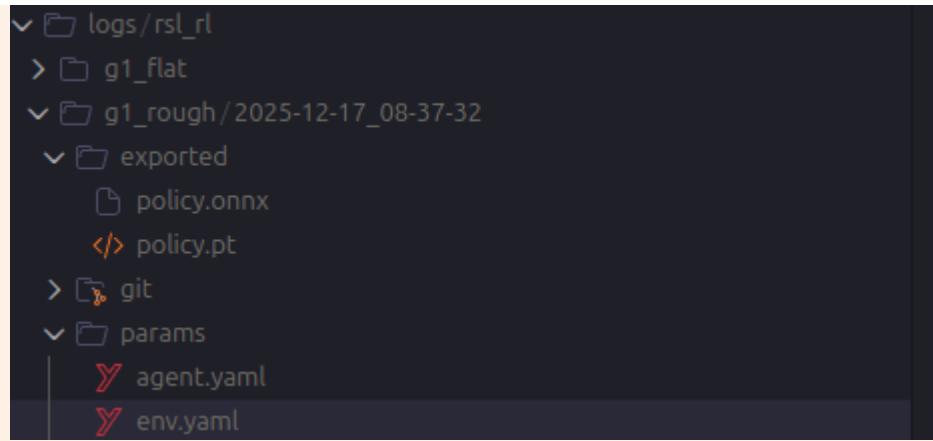


Al haber realizado el entrenamiento nos genera una política entrenada, ubicada dentro de la carpeta logs esa política puede ser exportada al Robot en físico y en simulación



⚠️ Importante

En la carpeta vista antes donde nos muestra la política para exportar, también hay una carpeta params dentro de la carpeta se guarda todos los parámetros del entorno usado para el entrenamiento, siendo mas específico el archivo env.yaml



- El contenido del archivo nos servirá para conocer la configuración, si se lleva la política a un entorno de simulación o real.