

**Robotika dan Sistem Cerdas**

***Webots Hacking***

Diajukan untuk memenuhi UAS pada mata kuliah  
Robotika dan Sistem Cerdas

Oleh :

Darry Pratama Putra (1103194056) (TK-43-GAB)



**Telkom  
University**

**PROGRAM STUDI S1 TEKNIK KOMPUTER**

**FAKULTAS TEKNIK ELEKTRO**

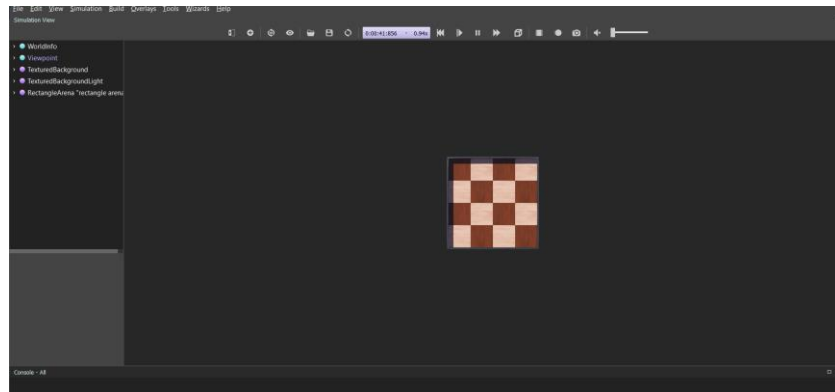
**UNIVERSITAS TELKOM**

**BANDUNG**

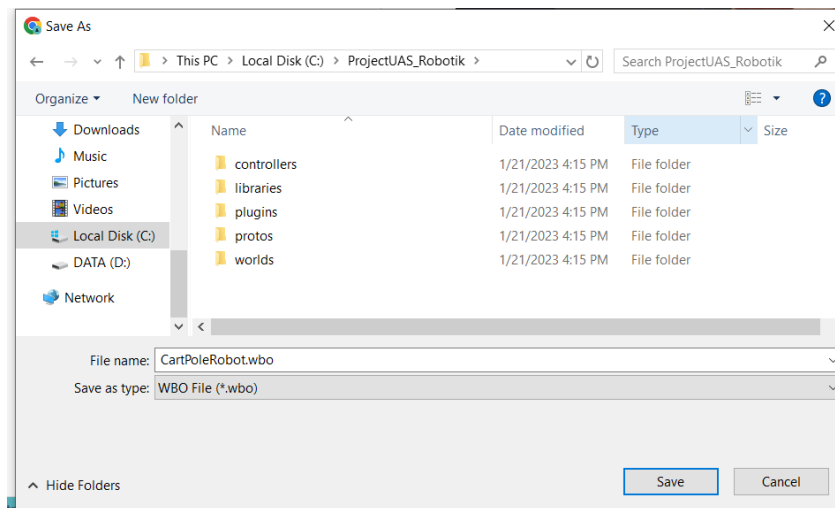
**2023**

# Deepbots-CartPole Beginner Tutorial

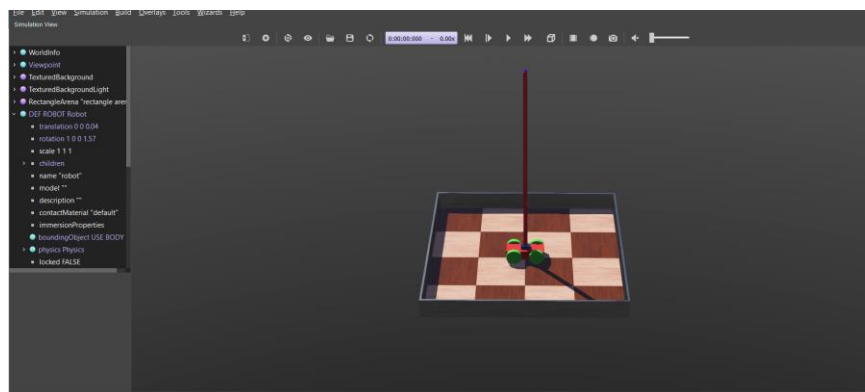
Menambahkan new Project Directory



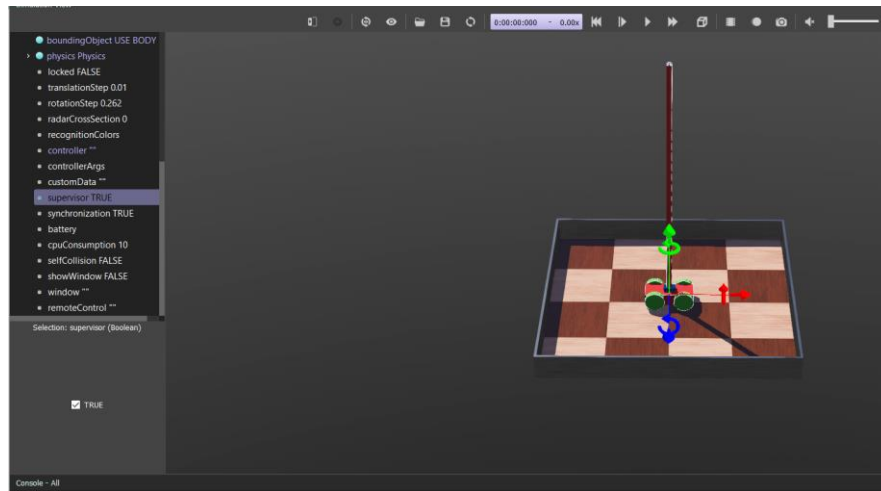
Klik link pada tutorial project, dan unduh content nya lalu simpan file tersebut.



Setelah itu import file untuk menampilkan robot seperti ini



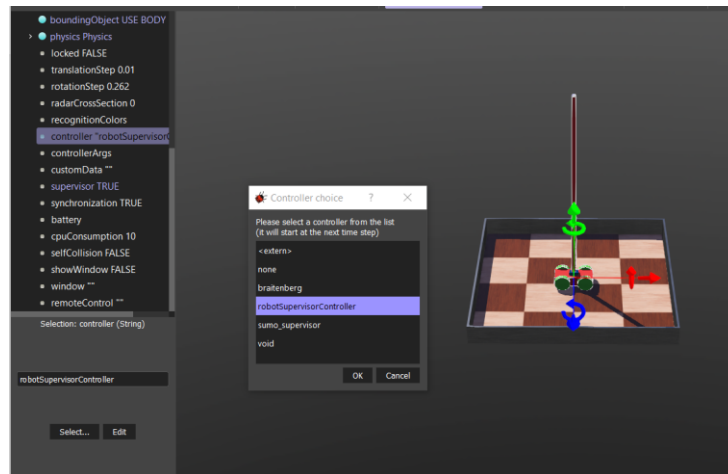
Lalu supervisor nya diubah menjadi TRUE.



Tambahkan controller baru

```
robotSupervisorController.py x
1 """robotSupervisorController controller."""
2
3 # You may need to import some classes of the
4 # from controller import Robot, Motor, Display
5 from controller import Robot
6
7 # create the Robot instance.
8 robot = Robot()
9
10 # get the time step of the current world.
11 timestep = int(robot.getBasicTimeStep())
12
13 # You should insert a getDevice-like function in the
14 # instance of a device of the robot. Something like:
15 # motor = robot.getDevice('motorname')
16 # ds = robot.getDevice('dsname')
17 # ds.enable(timestep)
18
19 # Main Loop:
20 # - perform simulation steps until Webots is finished
21 while robot.step(timestep) != -1:
22     # Read the sensors:
23     # Enter here functions to read sensor data
24     # val = ds.getValue()
25
26     # Process sensor data here.
27
28     # Enter here functions to send actuator commands to the robot
29     # motor.setPosition(10.0)
30     pass
31
32 # Enter here exit/destroy code.
33
```

Lalu tambahkan controller robotsupervisorController



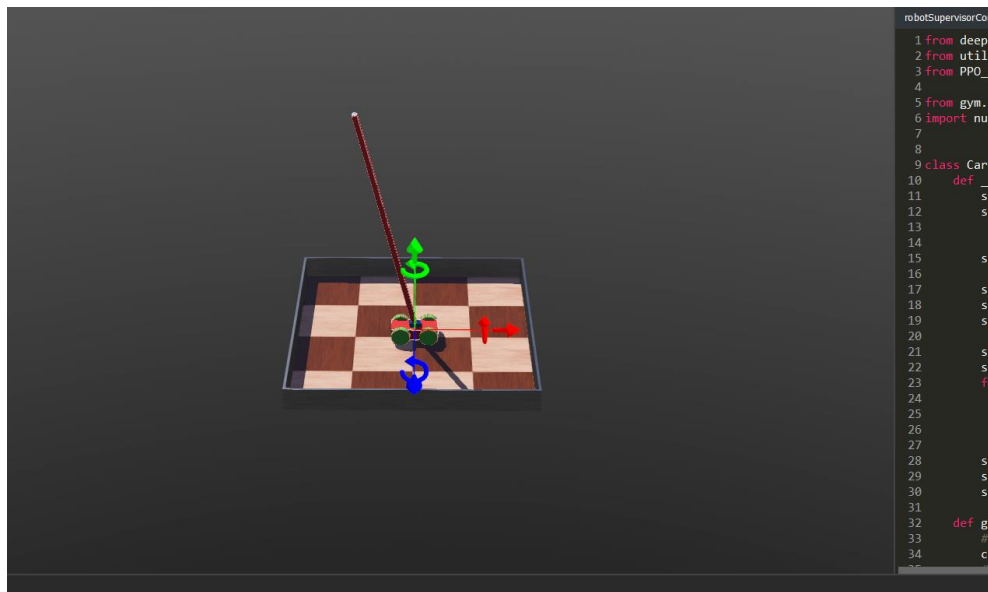
Selanjut save file yang sudah di download dengan tempat folder yang sama.

Name	Date modified	Type	Size
PPO_agent.py	1/21/2023 4:26 PM	Python File	8 KB
robotSupervisorController.py	1/21/2023 4:22 PM	Python File	1 KB
utilities.py	1/21/2023 4:36 PM	Python File	2 KB

Lalu Copy and paste code ke halaman yang tadi.

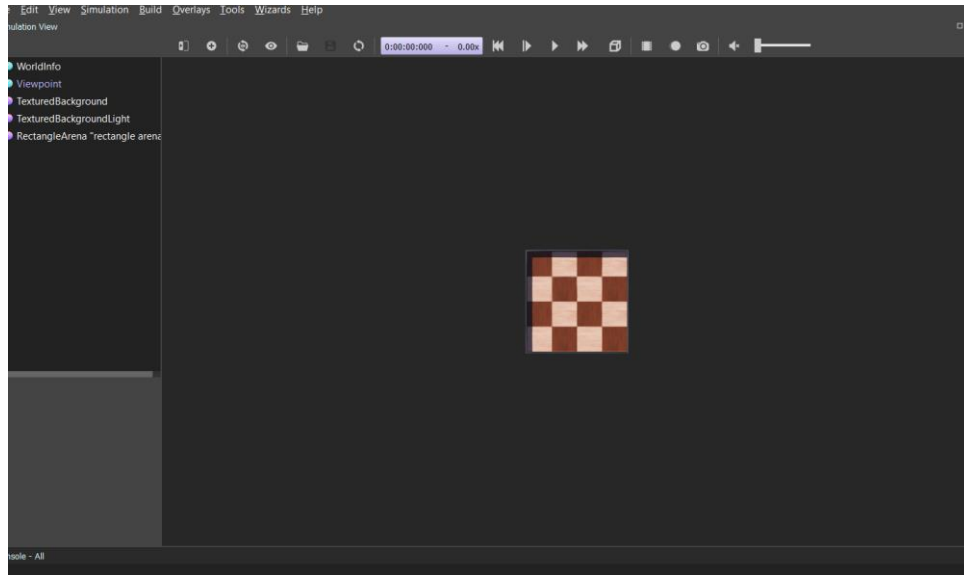
```
1 from deepbots.supervisor.controllers.robot import RobotSupervisor
2 from utilities import normalizeToRange, positionToAngle
3 from PPO_agent import PPOAgent, Transition
4
5 from gym.spaces import Box, Discrete
6 import numpy as np
7
8
9 class CartpoleRobot(RobotSupervisor):
10     def __init__(self):
11         super().__init__()
12         self.observation_space = Box(low=-1.5, high=1.5, dtype=np.float32)
13         self.action_space = Discrete(2)
14
15         self.robot = self.getSelf() # Get the robot
16         self.positionSensor = self.getDevice('positionSensor')
17         self.positionSensor.enable(self.task)
18
19         self.poleEndpoint = self.getFromDevice('poleEndpoint')
20         self.wheels = []
21         for wheelName in ['wheel1', 'wheel2']:
22             wheel = self.getDevice(wheelName)
23             wheel.setPosition(float('inf'))
24             wheel.setVelocity(0.0) # Zero velocity
25             self.wheels.append(wheel)
26
27         self.stepsPerEpisode = 200 # Max steps per episode
28         self.episodeScore = 0 # Score achieved in this episode
29         self.episodeScoreList = [] # All episode scores
30
31     def get_observations(self):
32         # Position on x axis
33         cartPosition = normalizeToRange(self.robot.getPosition(), self.observation_space)
34         # Pole angle in radians
```

Robot telah berhasil bergerak.

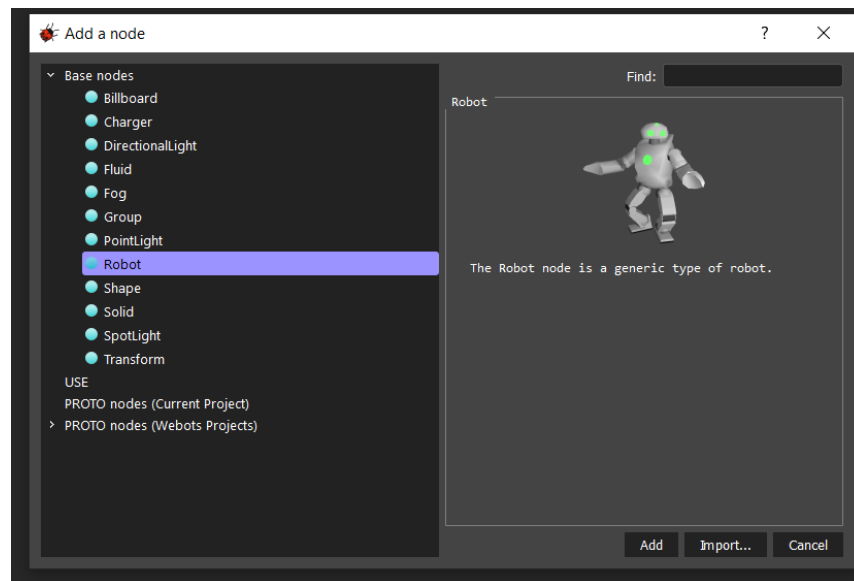


# *CartPole Emitter-Receiver Scheme Tutorial*

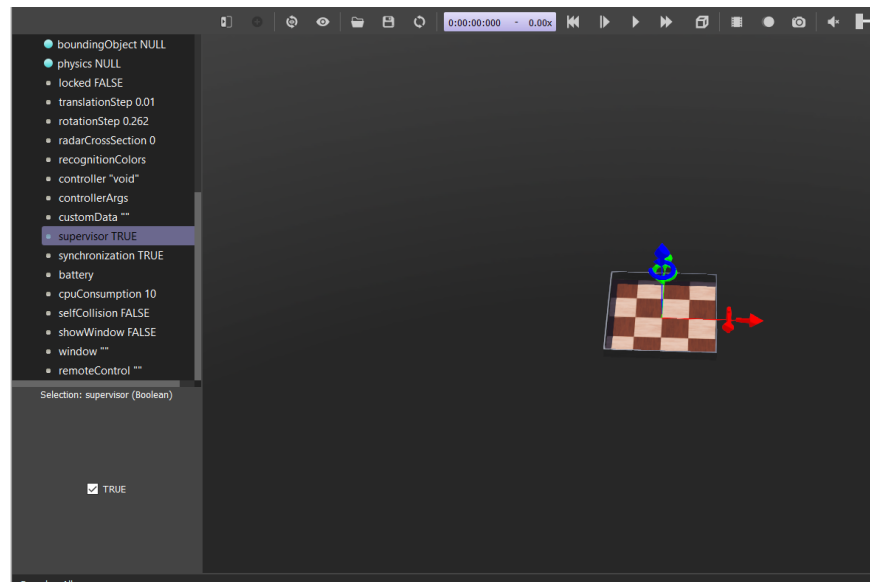
Membuat Project baru pada tab wizard, Create New Project



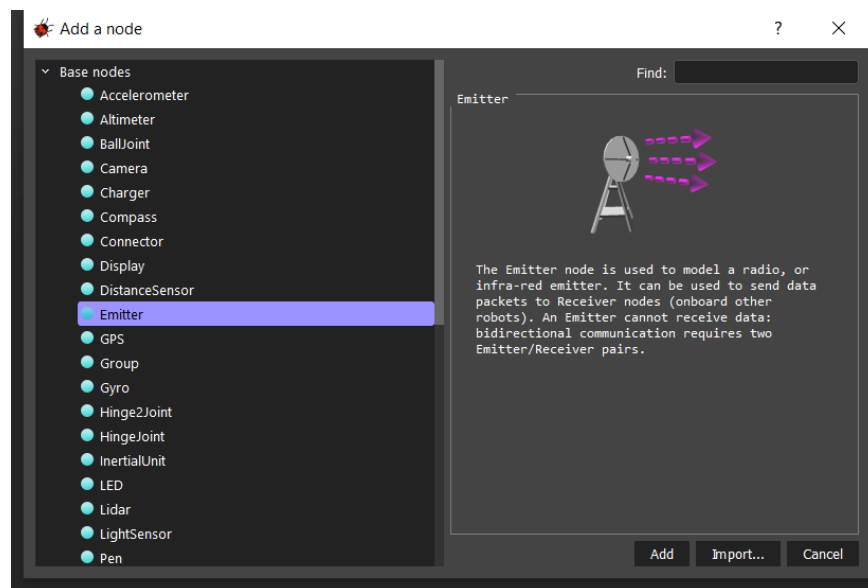
Pilih Sub Base Nodes, lalu import Robot



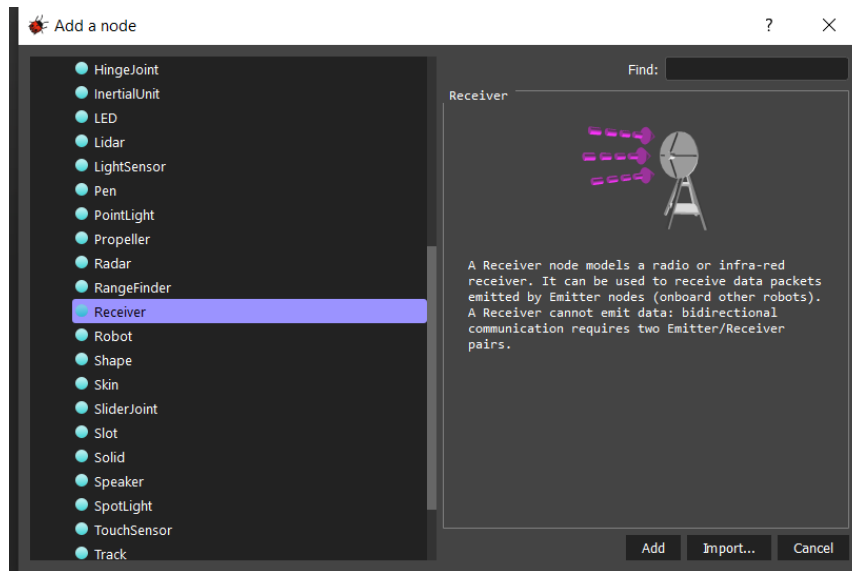
Lalu pilih Supervisor yang awal nya FALSE, menjadi TRUE.



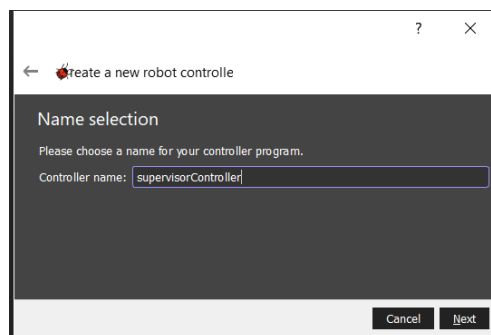
Lalu Pilih Base nodes untuk menambahkan Emitter



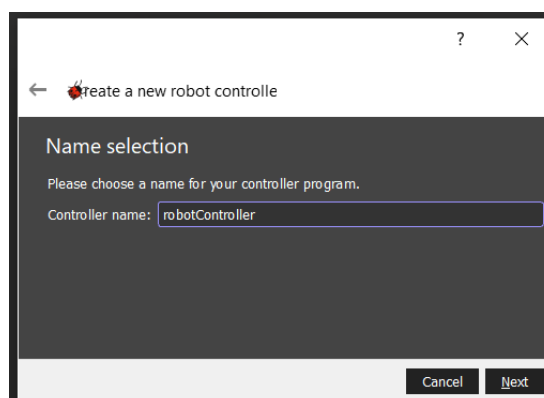
Lalu lanjut ke step berikutnya, menambahkan Receiver lalu klik save



Selanjutnya menambahkan Controllers

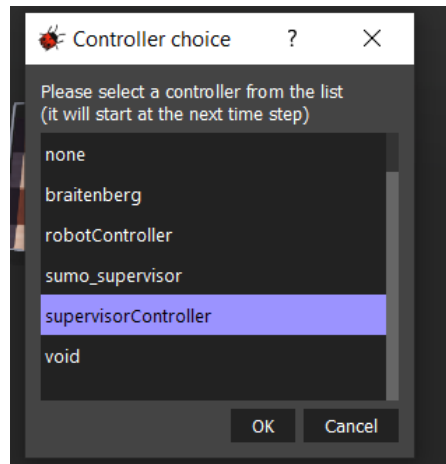


Lalu merubah controller name menjadi robotController





Asosiasikan controller yang telah anda buat dengan field controller, lalu pilih SupervisorController dan Klik save



Simpan file yang telah di unduh dengan folder yang sama, dan folder berbeda pada project sebelum nya

Name	Date modified	Type	Size
PPOAgent.py	1/21/2023 4:54 PM	Python File	8 KB
supervisorController.py	1/21/2023 4:50 PM	Python File	1 KB
utilities.py	1/21/2023 4:54 PM	Python File	1 KB

Selanjutnya menambahkan code pada robotcontroller. Code bisa di unduh dari tutorial yang telah disediakan, lalu copy and paste pada laman tersebut dan save.

```
supervisorController.py x robotController.py x
1 import numpy as np
2 from deepbots.supervisor.controllers.supervisor import SupervisorCSV
3 from PPOAgent import PPOAgent, Transition
4 from utilities import normalizeToRange
5
6
7 class CartPoleSupervisor(SupervisorCSV):
8     def __init__(self):
9         super().__init__()
10        self.observationSpace = 4
11        self.actionSpace = 2
12
13        self.robot = None
14        self.respawnRobot()
15        self.poleEndpoint = 0.0
16        self.messageReceived = False
17
18        self.episodeCount = 0
19        self.episodeLimit = 1000
20        self.stepsPerEpisode = 1000
21        self.episodeScore = 0
22        self.episodeScoreLimit = 1000
23
24    def respawnRobot(self):
25        if self.robot is not None:
26            # Despawn robot
27            self.robot.destroy()
28
29            # Respawn robot in the center
30            rootNode = self.supervisor.getRootNode()
31            childrenField = rootNode.getChildrenField()
```

```

supervisorController.py  x  robotController.py  x
1 from deepbots.robots.controllers.robot_emi
2
3
4 class CartpoleRobot(RobotEmitterReceiverCS
5     def __init__(self):
6         super().__init__()
7         self.positionSensor
8         self.positionSensor.
9         self.wheel1 = self.r
10        self.wheel1.setPosit
11        self.wheel1.setVeloc
12        self.wheel2 = self.r
13        self.wheel2.setPosit
14        self.wheel2.setVeloc
15        self.wheel3 = self.r
16        self.wheel3.setPosit
17        self.wheel3.setVeloc
18        self.wheel4 = self.r
19        self.wheel4.setPosit
20        self.wheel4.setVeloc
21
22    def create_message(self):
23        # Read the sensor va
24        message = [str(self.
25        return message
26
27    def use_message_data(self, mess
28        action = int(message
29
30        if action == 0:
31            motorSpee
32

```

Setelah itu klik run pada webots untuk menjalankan sistem tersebut.

