Topic	HUMANOID - UNDERSTANDING MOTION FILES	
Class Description	**Students will start working with a humanoid robot. They will understand what motion files are and will learn to use pre-written motion files with their robots.**	
Class	**PRO C295**	
Class time	**45 mins**	
Goal	• Adding a humanoid robot • Understanding motion files • Learn to code movement of the robot using motion files	

Resources Required	• Teacher Resources:
	  ○ Laptop with internet connectivity
	  ○ Earphones with mic
	  ○ Notebook and pen
	  ○ Smartphone
	• Student Resources:
	  ○ Laptop with internet connectivity
	  ○ Earphones with mic
	  ○ Notebook and pen	

Class structure	**Warm-Up** **Teacher-Led Activity 1** **Student-Led Activity 1** **Wrap-Up**	**10 mins** **10 mins** **20 mins** **05 mins**
Credit & Permissions:	**This project uses Webots, an open-source mobile robot simulation software developed by Cyberbotics Ltd.** **License**	
WARM-UP SESSION - 10 mins		

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

Teacher Action	Student Action
Hey <student's name>. How are you? It's great to see you! Are you excited to learn something new today? **Following are the WARM-UP session deliverables:** • Greet the student. • Revision of previous class activities. • Quizzes.	**ESR**: Hi, thanks! Yes I am excited about it! Click on the slide show tab and present the slides
WARM-UP QUIZ Click on In-Class Quiz	
Activity Details **Following are the session deliverables:** • Appreciate the student. • Narrate the story by using hand gestures and voice modulation methods to bring in more interest in students.	
TEACHER-LED ACTIVITY - 10 mins	
Teacher Initiates Screen Share	
• **Adding the humanoid.** • **Understanding how to use the motion files.**	
Teacher Action	**Student Action**
So what did we learn in our last class? *The teacher will clarify if there are any doubts!* So let's get started with today's class.	**ESR:** Varied!

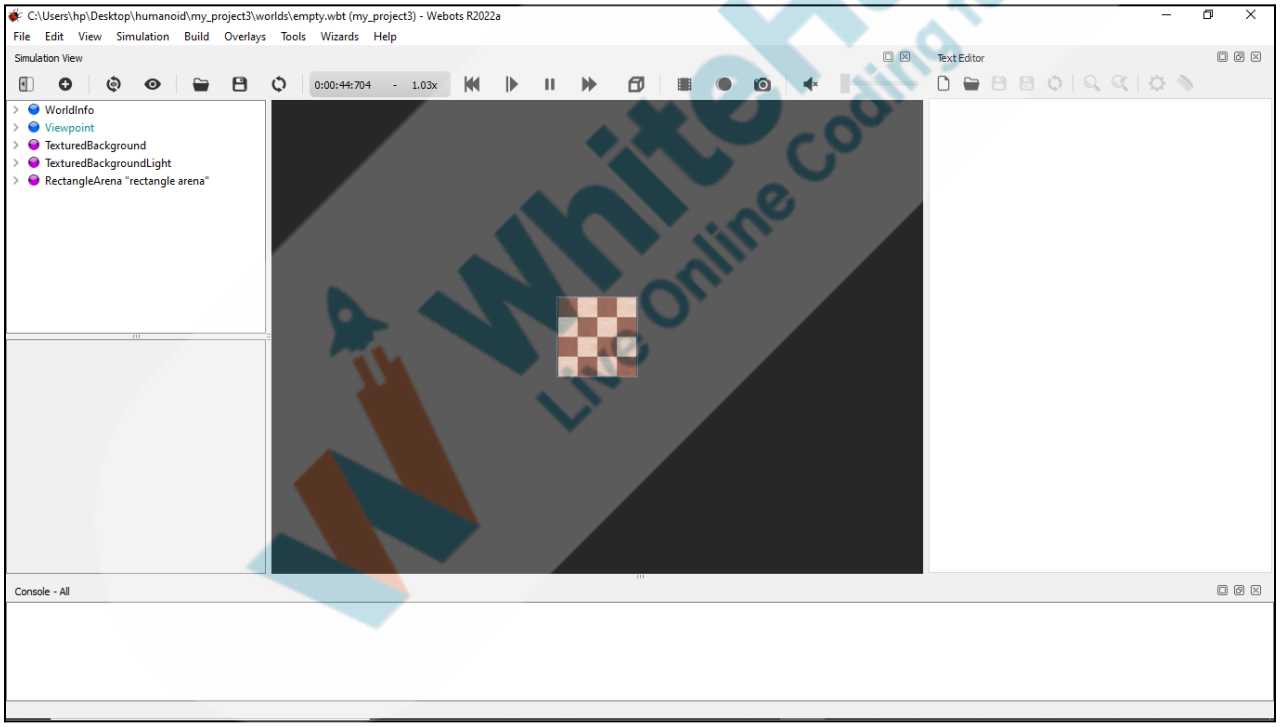
© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

Let's get started with a new project.

1. Open **Webots** Robot Simulator.
2. Go to **File** in the menu bar → click on **New World**
3. Go to **Wizards** in the menu bar → click on **New Project Directory…** (Add a rectangular arena while creating the project)

Note: Make sure you save the project after every major step. Otherwise, you might lose your work.

This is how your screen should look at this point.

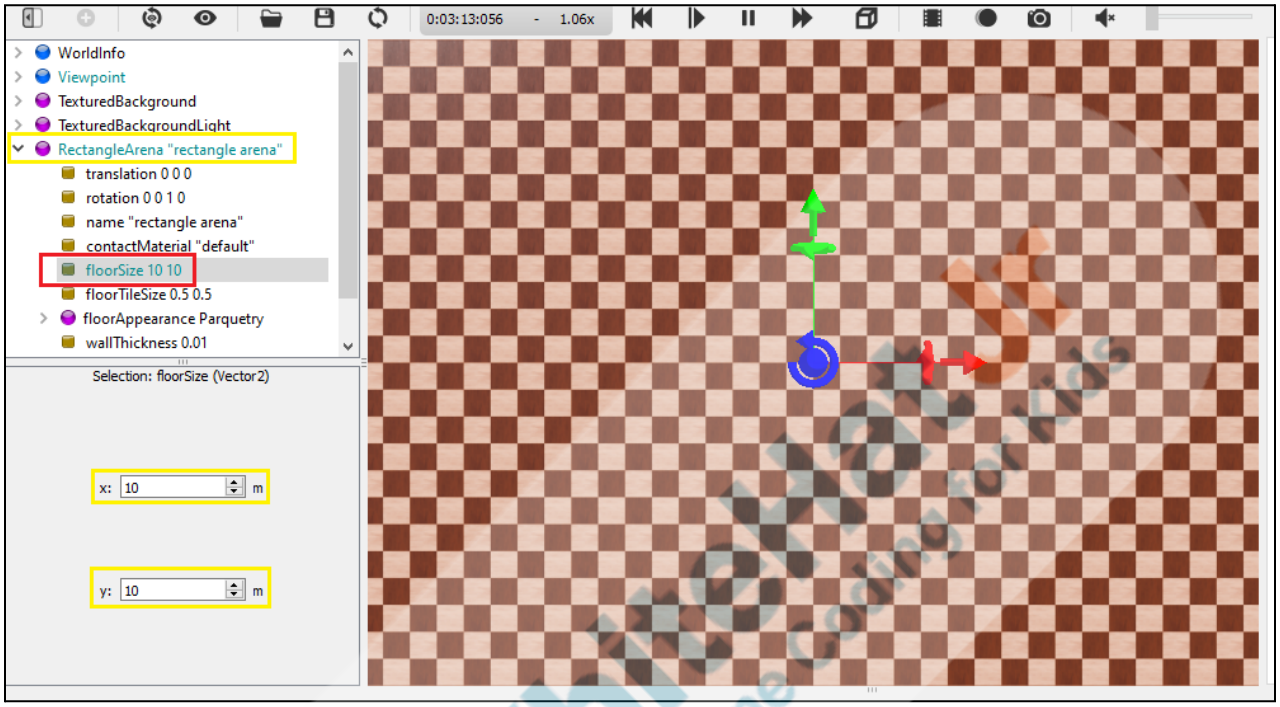


Now, what should we do?

ESR: Let's increase the floor size.

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

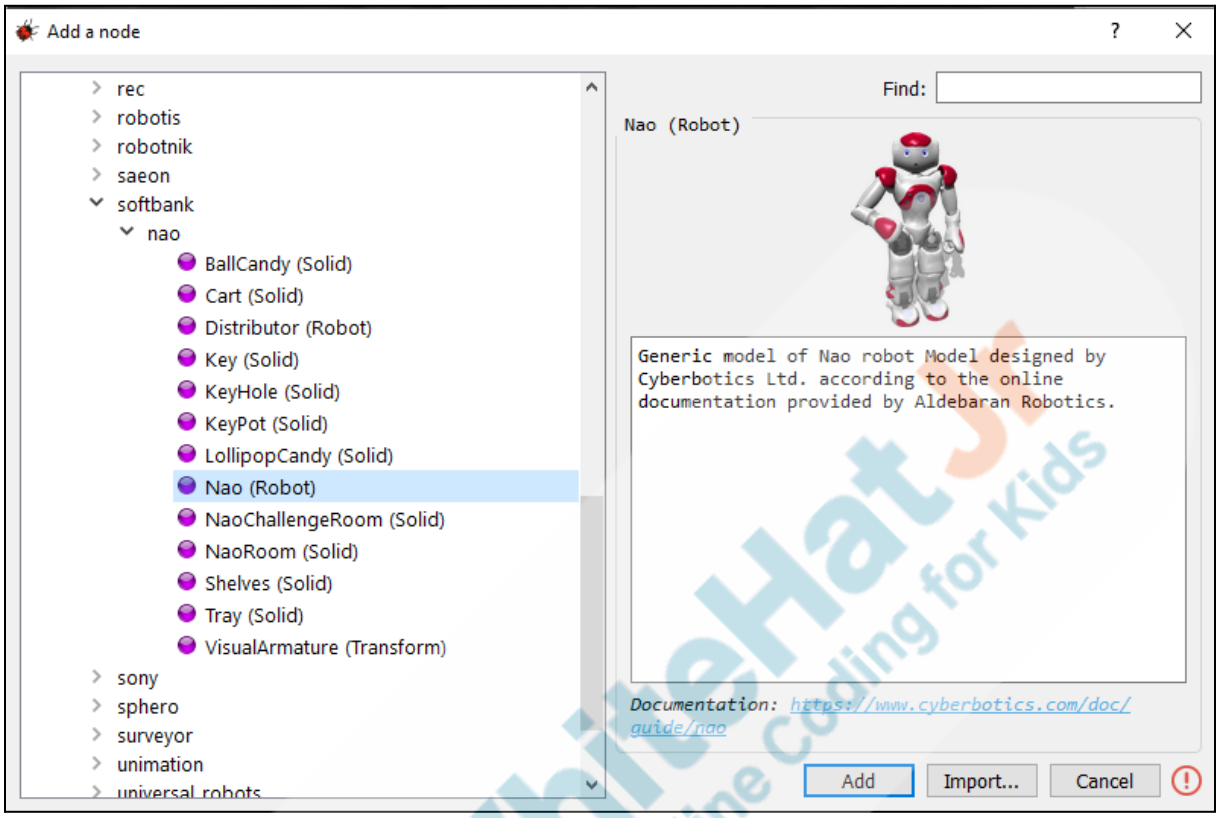
Yes. Let's increase it to 10x10.



Now, we need to add a robot. Let's add the humanoid named **Nao**.

1. Click on the Add New  button.
2. Go to **PROTO nodes (Webots Projects)** → **robots** → **softbank** → **nao** → **Nao(Robot)**

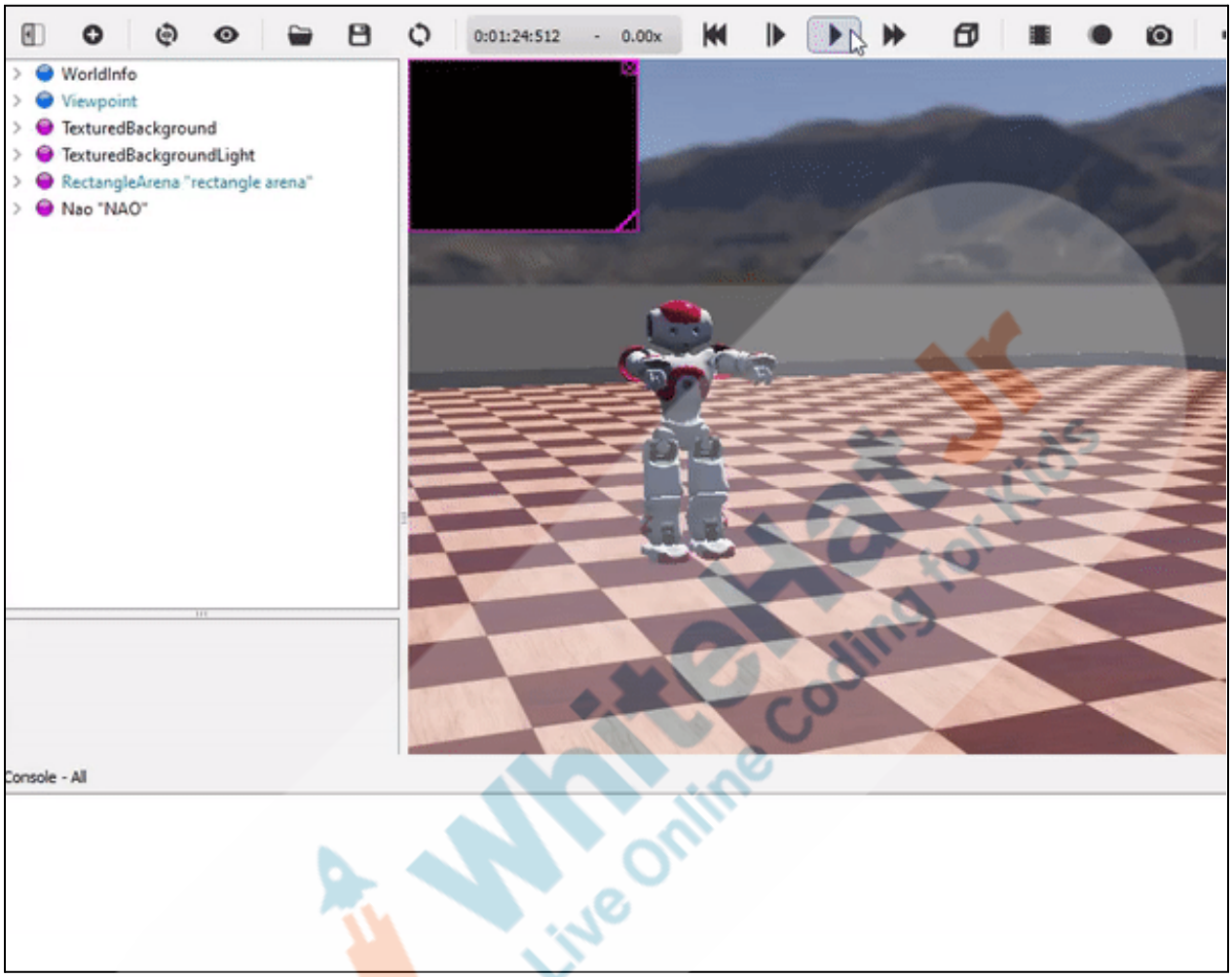
© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



This robot already has some pre-written controllers. Let's run it once.

The teacher runs the simulation and presses different arrow keys to show the movement/ functionalities of the Nao robot.

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

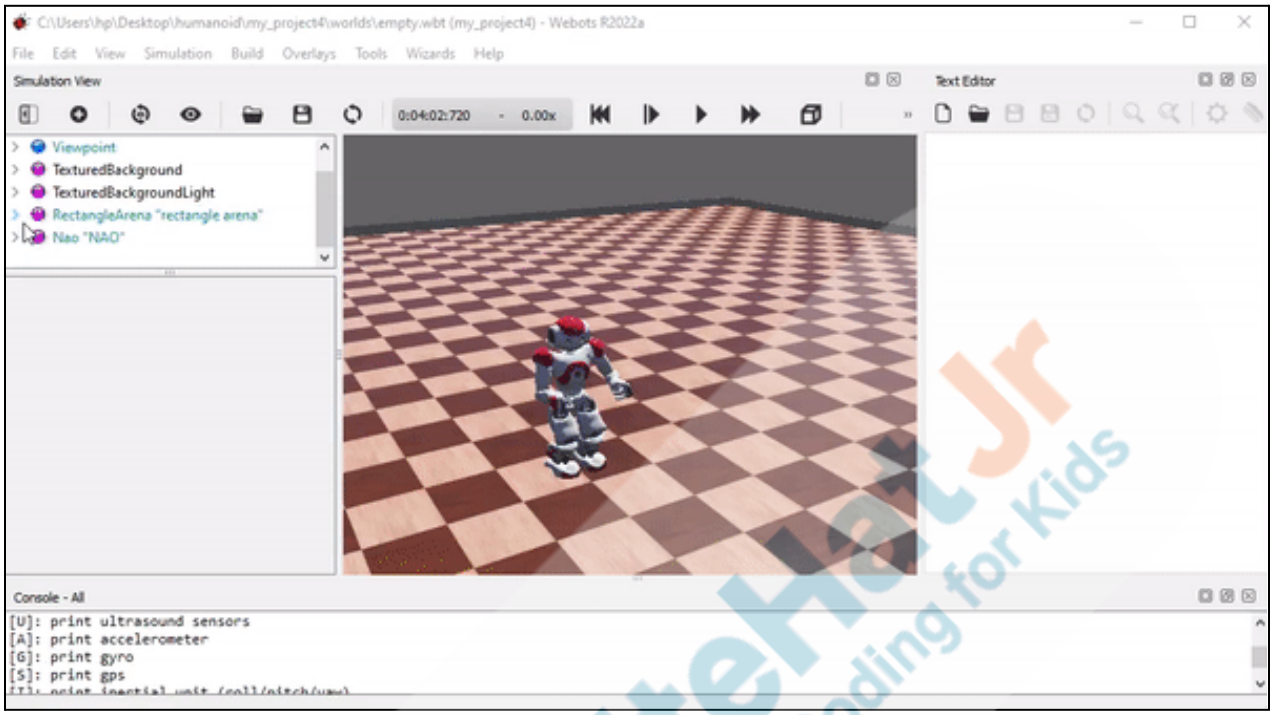


Click here to view the reference video.

We can look at the pre-written controller to understand how the code was written.

1. Click on the **Nao robot** → go to the **controller** property → click on **Select…** → change the controller to **nao_demo_python.** (As we want to observe the python code)
2. To observe the code, click on **edit**.

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



Click here to view the reference video.

3. Let's observe the code. We can see that there's a lot of code written in this file. The good thing is that it is somewhat comprehensible to us.

   I want you to focus on the very first part.

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

```
14
15 """Example of Python controller for Nao robot.
16    This demonstrates how to access sensors and actuators"""
17
18 from controller import Robot, Keyboard, Motion
19
20
21 class Nao (Robot):
22     PHALANX_MAX = 8
23
24     # load motion files
25     def loadMotionFiles(self):
26         self.handWave = Motion('../../motions/HandWave.motion')
27         self.forwards = Motion('../../motions/Forwards50.motion')
28         self.backwards = Motion('../../motions/Backwards.motion')
29         self.sideStepLeft = Motion('../../motions/SideStepLeft.motion')
30         self.sideStepRight = Motion('../../motions/SideStepRight.motion')
31         self.turnLeft60 = Motion('../../motions/TurnLeft60.motion')
32         self.turnRight60 = Motion('../../motions/TurnRight60.motion')
33
```

4. We can see that they are using a new class called **Motion.**

```
18 from controller import Robot, Keyboard, Motion
19
20
21 class Nao (Robot):
22     PHALANX_MAX = 8
23
24     # load motion files
25     def loadMotionFiles(self):
26         self.handWave = Motion('../../motions/HandWave.mo
27         self.forwards = Motion('../../motions/Forwards50.
```

5. If we observe the next portion of code, we have a **loadMotionFiles()** method.

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

```
from controller import Robot, Keyboard, Motion


class Nao (Robot):
    PHALANX_MAX = 8

    # load motion files
    def loadMotionFiles(self):
        self.handWave = Motion('../../motions/HandWave.motion')
        self.forwards = Motion('../../motions/Forwards50.motion')
        self.backwards = Motion('../../motions/Backwards.motion')
        self.sideStepLeft = Motion('../../motions/SideStepLeft.motion')
        self.sideStepRight = Motion('../../motions/SideStepRight.motion')
        self.turnLeft60 = Motion('../../motions/TurnLeft60.motion')
        self.turnRight60 = Motion('../../motions/TurnRight60.motion')

    def startMotion(self, motion):
        # interrupt current motion
        if self.currentlyPlaying:
            self.currentlyPlaying.stop()
```

Let's understand what these **.motion** files are.

Motion files specify motion sequences that usually involve several motors playing simultaneously, e.g., a walking sequence, a HandWave sequence, etc.

Motion class has functions using which we can load, play, stop these **.motion** files.

But where can we find these motion files?

ESR: In the project folder.

Let's try to find these files and let's dive deeper.

Make sure you do not change anything in the following files and folders.

For Windows Operating System :

Go to "**C:**" drive → double click on **Program Files** → go to **Webots** folder → then go to **projects** folder → **robots** folder → **softbank** folder→ **nao** folder → **motions** folder

For MAC Operating System :

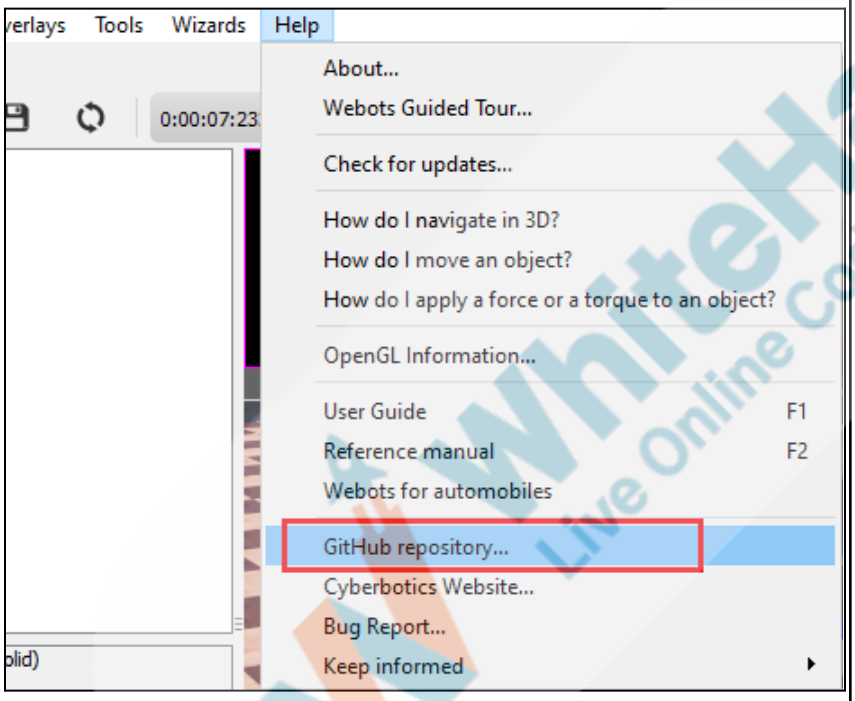
Go to "**Finder**" → click on **Applications** → right

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

click on **Webots** application → select "**Show Package Contents**" → open **projects** folder→ **robots** folder → **softbank** folder→ **nao** folder → **motions** folder

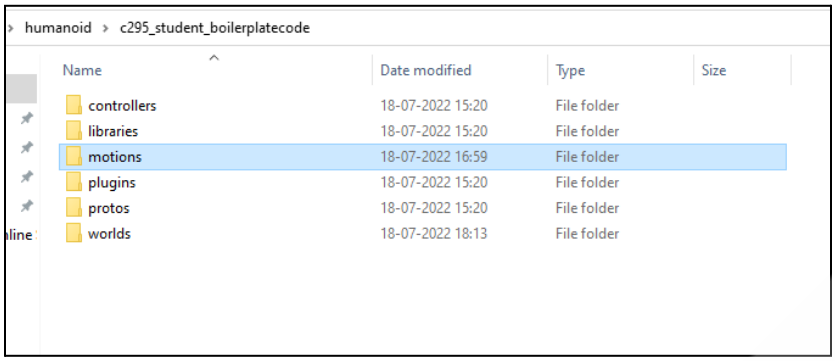
Alternate method:

Click on **Help** in the menu bar → go to **GitHub repository**… → open **projects** → **robots** folder → **softbank** folder→ **nao** folder → **motions** folder



6. Copy the **motions** folder and paste it in your current project directory so that we can use these files in our current project.

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

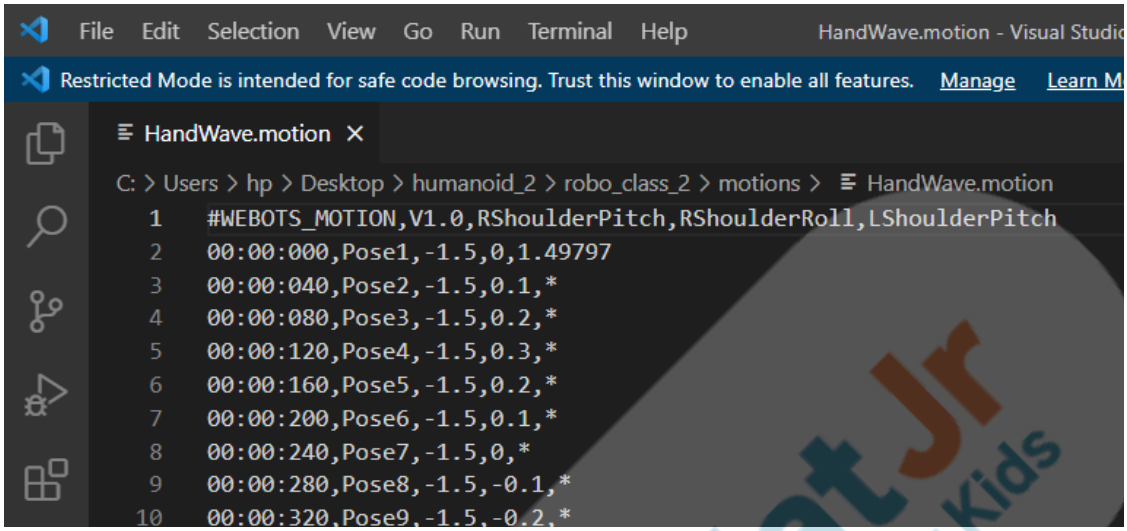
<table>
<tr><td> 
</td><td></td></tr>
<tr><td>Let’s open one of these files and see what it consists of. What do you think?

We can open this file with an editor. In this case, let’s open it with Visual Studio Code.

A hand wave is a relatively simpler motion, so let’s open that file first.

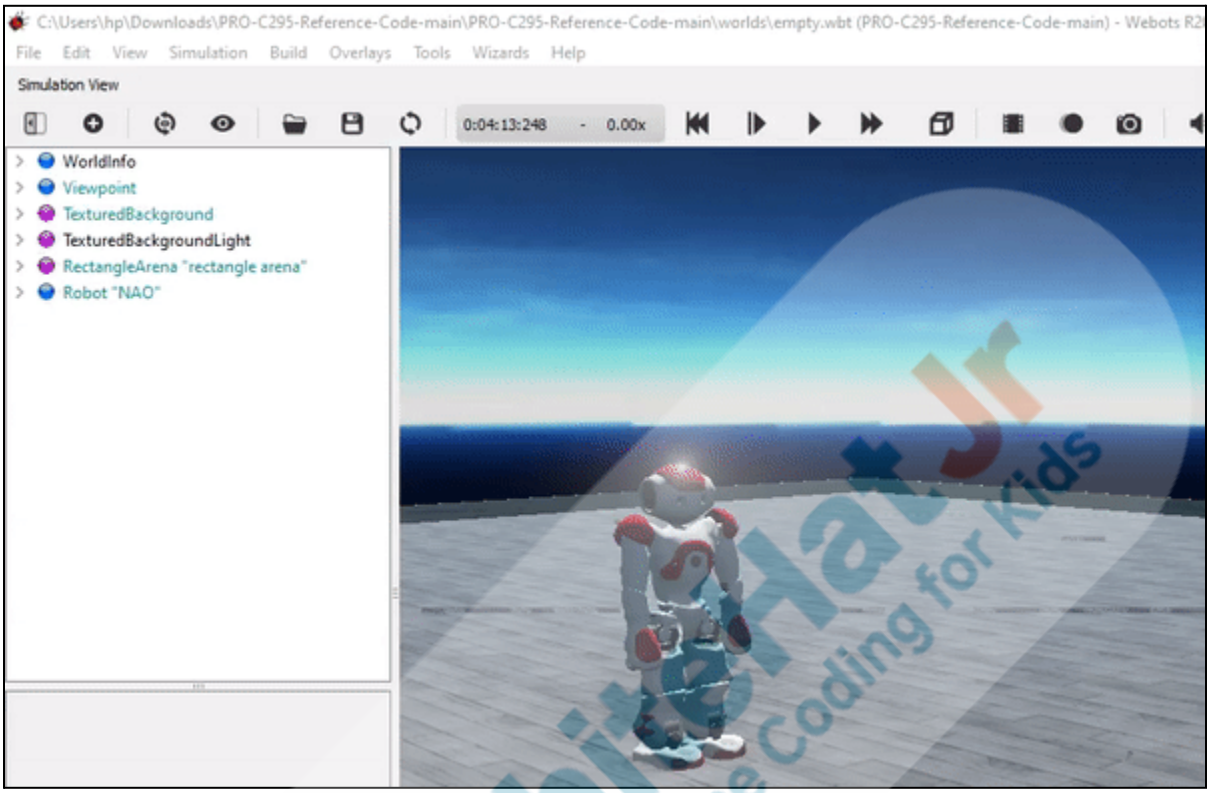
The teacher opens the HandWave.motion file. It can be found in the motion folder inside the project folder.</td><td>ESR: Sure. That would be great.</td></tr>
</table>

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



What do you think?	**ESR:** It is a really large file with a lot of numbers.
Yes. It is. But it will be easy to understand. Observe the first row. It consists of some names like - **RShoulderPitch, RShoulderRoll, LShoulderPitch**. What do you think these names represent?	**ESR:** They represent some components in the robot.
Correct. These are some components in the robot indeed. Let's go back to our robot in webots.	

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



Click here to view the reference video.

We can see that these are names of motors positioned to operate different joints of the robot.

Now, let's go back to the motion document.

1. The first column in this document represents the time. So, **00:00:000** represents the time when the motion begins. It is written in [mm:ss:fff] format where m = minute, s = second and f = milliseconds. So, if the value is **00:01:000**, it means that particular row of motion will be completed after a second.

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

```
#WEBOTS_MOTION V1.0,RShoulderPitch,RShoulderRoll,LShoulderPitch
00:00:000,Pose1,-1.5,0,1.49797
00:00:040,Pose2,-1.5,0.1,*
00:00:080,Pose3,-1.5,0.2,*
00:00:120,Pose4,-1.5,0.3,*
00:00:160,Pose5,-1.5,0.2,*
00:00:200,Pose6,-1.5,0.1,*
00:00:240,Pose7,-1.5,0,*
00:00:280,Pose8,-1.5,-0.1,*
00:00:320,Pose9,-1.5,-0.2,*
00:00:360,Pose10,-1.5,-0.3,*
00:00:400,Pose11,-1.5,-0.4,*
00:00:440,Pose12,-1.5,-0.5,*
00:00:480,Pose13,-1.5,-0.5,*
00:00:520,Pose14,-1.5,-0.4,*
00:00:560,Pose15,-1.5,-0.3,*
00:00:600,Pose16,-1.5,-0.2,*
```

2. The second column represents the name of the poses.

```
#WEBOTS_MOTION,V1.0,RShoulderPitch,RShoulderRoll,LShoulderPitch
00:00:000,Pose1,-1.5,0,1.49797
00:00:040,Pose2,-1.5,0.1,*
00:00:080,Pose3,-1.5,0.2,*
00:00:120,Pose4,-1.5,0.3,*
00:00:160,Pose5,-1.5,0.2,*
00:00:200,Pose6,-1.5,0.1,*
00:00:240,Pose7,-1.5,0,*
00:00:280,Pose8,-1.5,-0.1,*
00:00:320,Pose9,-1.5,-0.2,*
00:00:360,Pose10,-1.5,-0.3,*
00:00:400,Pose11,-1.5,-0.4,*
00:00:440,Pose12,-1.5,-0.5,*
00:00:480,Pose13,-1.5,-0.5,*
00:00:520,Pose14,-1.5,-0.4,*
00:00:560,Pose15,-1.5,-0.3,*
00:00:600,Pose16,-1.5,-0.2,*
```

3. We pass the **position** value of the given motor from the 3rd column onwards.

   Previously, we assigned these positions to the motors by using the **setPosition()**

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

<table>
<tr><td colspan="2">method.

By using motion files, we can do the same thing much more easily.</td></tr>
<tr><td>Now, you will write your own controller for the Nao robot with these motion files that we have just discovered.</td><td></td></tr>
<tr><td>Are you ready?</td><td>ESR: Yes.</td></tr>
<tr><td colspan="2">Teacher Stops Screen Share</td></tr>
<tr><td>So now it’s your turn.
Please share your screen with me.</td><td></td></tr>
<tr><td>We have one more class challenge for you.
Can you solve it?

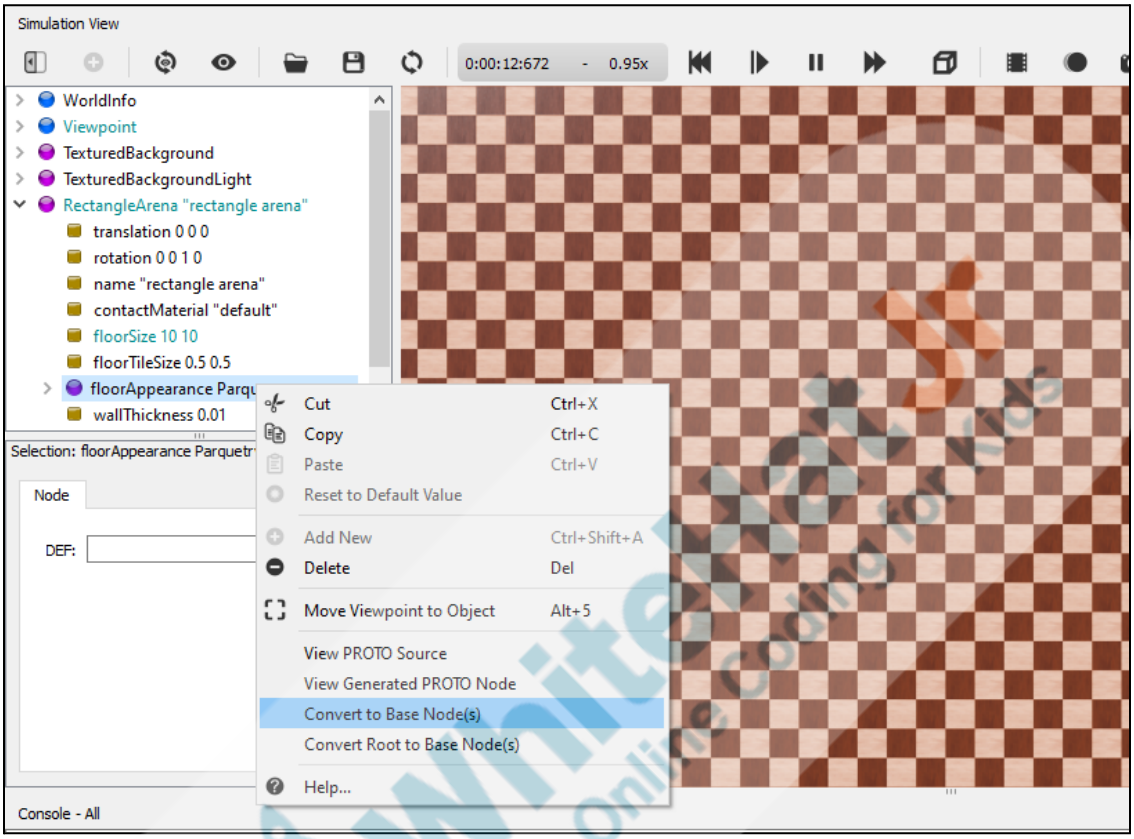
Let's try. I will guide you through it.</td><td></td></tr>
<tr><td colspan="2">STUDENT-LED ACTIVITY - 20 mins</td></tr>
<tr><td colspan="2">• Ask the student to press the ESC key to come back to the panel.
• Guide the student to start Screen Share.
• The teacher gets into Full Screen.</td></tr>
<tr><td colspan="2">Student Initiates Screen Share</td></tr>
<tr><td colspan="2"><u>ACTIVITY</u>

• Code to play the .motion files on your robot.</td></tr>
<tr><td>Teacher Action</td><td>Student Action</td></tr>
<tr><td>Student downloads the boilerplate code from Student Activity 1.</td><td></td></tr>
<tr><td colspan="2">Let’s set up our robot’s environment first. Change the floor texture and background lights.

1. To change the floor texture:
a. Go to scene panel → click RectangleArena → right click on</td></tr>
</table>

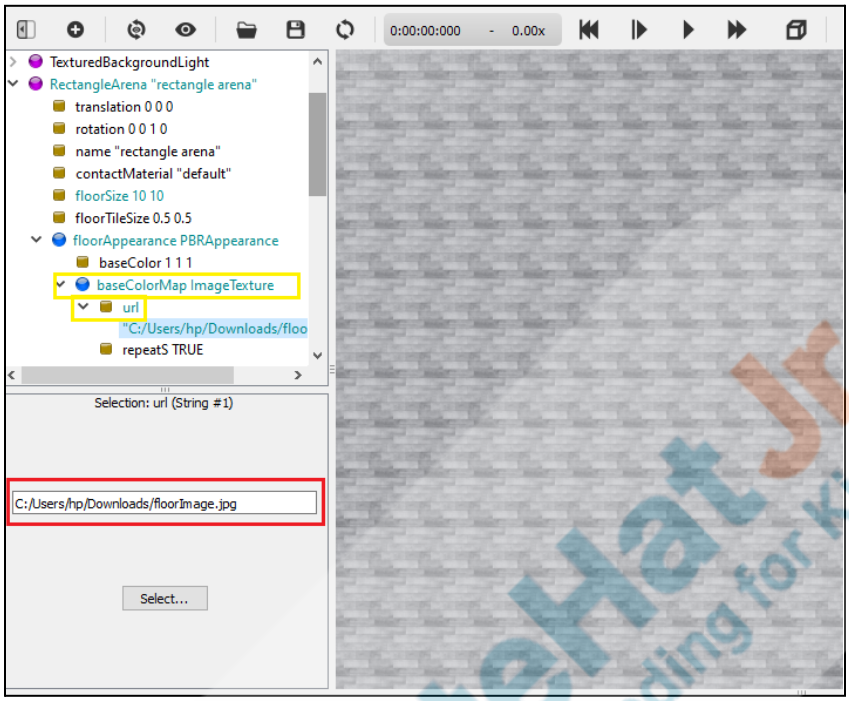
© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

floorAppearance → click on **Convert to Base Node(s)**



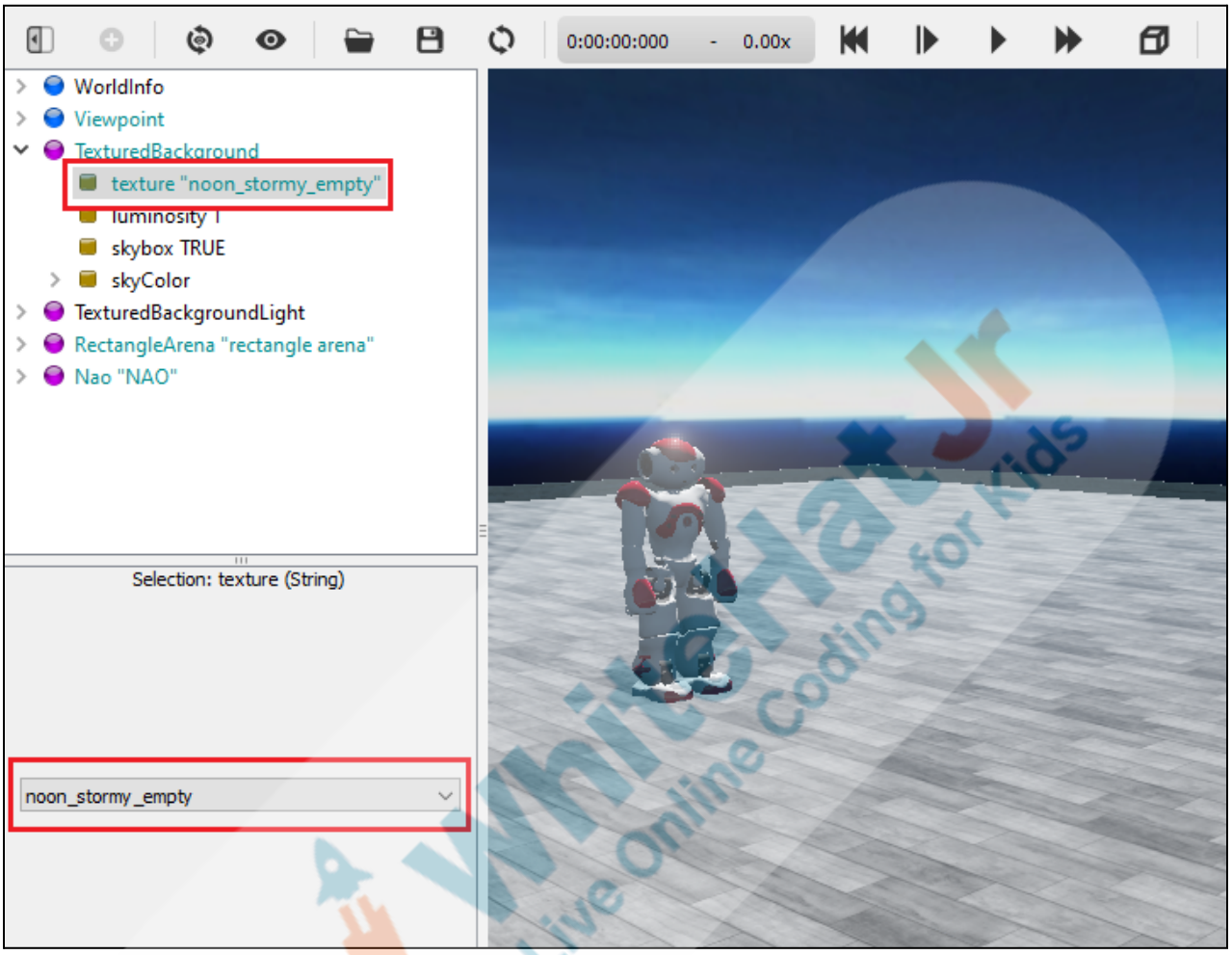
b. Download the floor image from Student Activity 2
c. Now, expand the **floorAppearance** node. Find **baseColorMap ImageTexture** and go to the **url property.** Here, select the downloaded image as the texture.

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



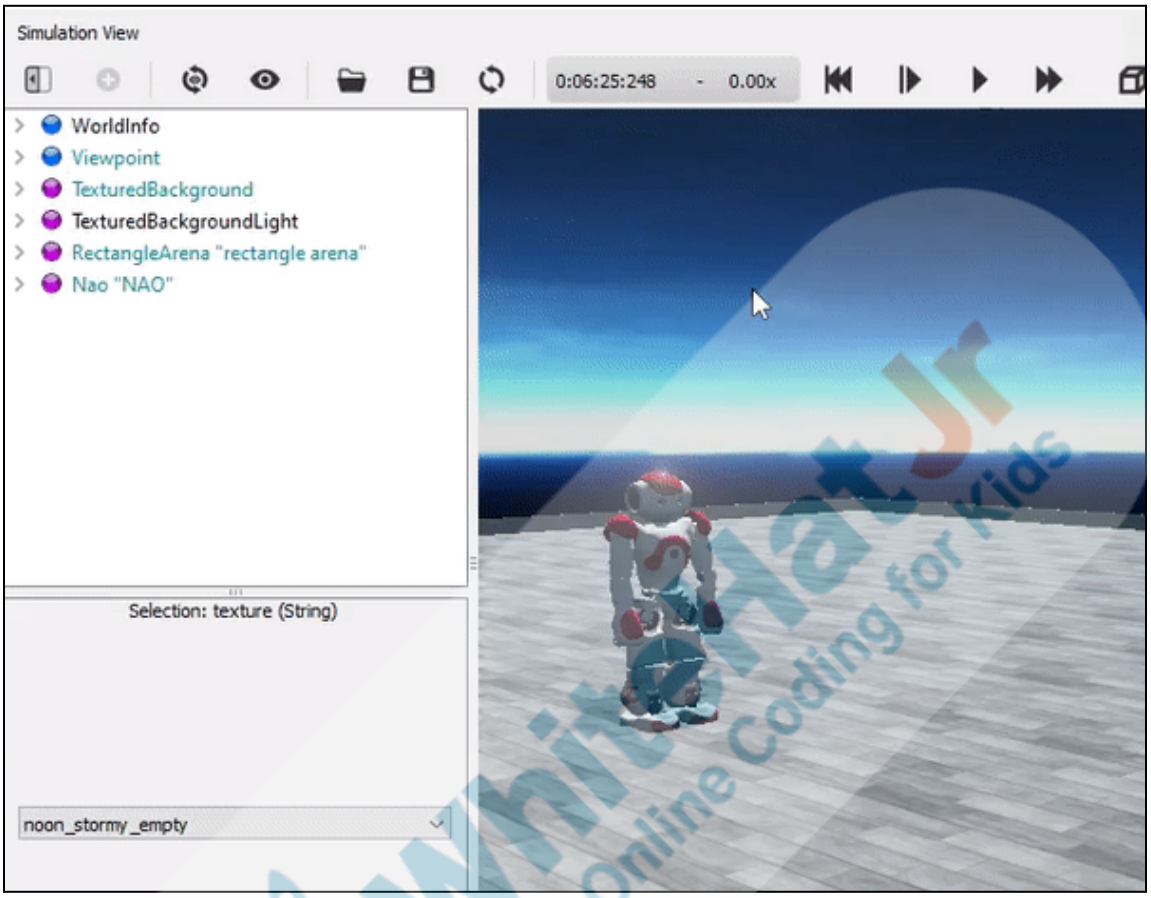
Now, let's change the TexturedBackground as well.

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



Now, what should we do? Sure! To do that we need to convert the Nao robot to a **Base node(s).** 1. Right click on **Nao** robot → click on **Convert to Base Node(s)**	**ESR:** We need to write the controller for the robot.

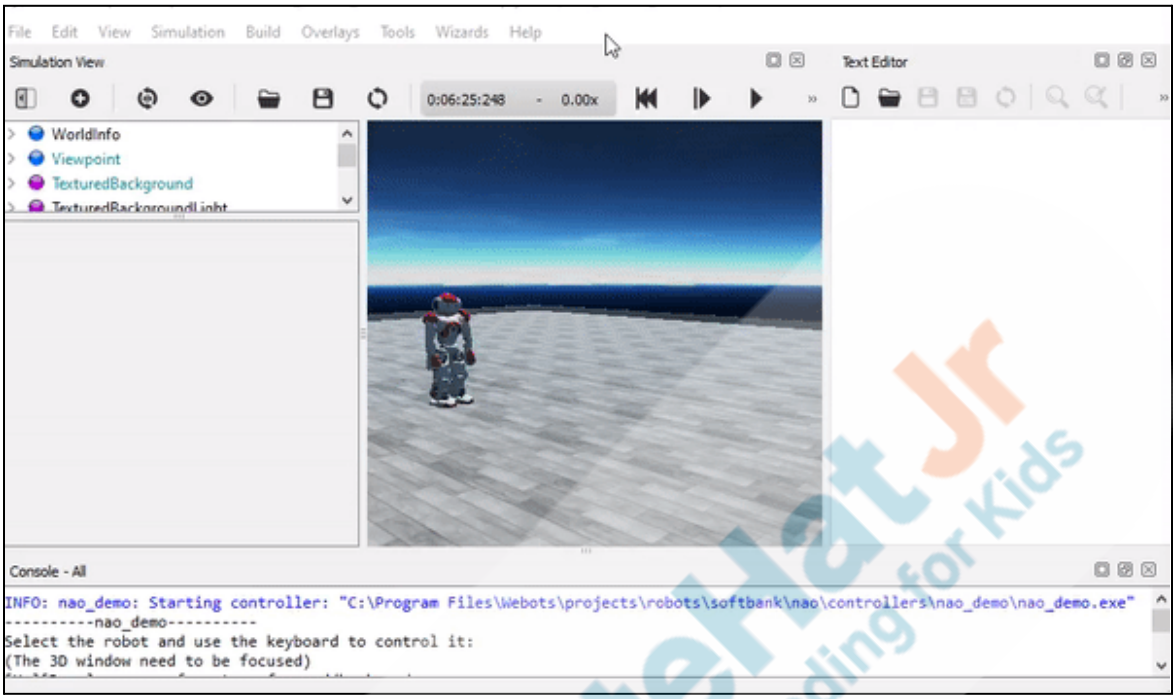
© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



Click here to view the reference video.

2. After that, create a **new project controller.** Go to **Wizards** in the menu bar → click on **New Robot Controller…** → name it as **robo_controller.**

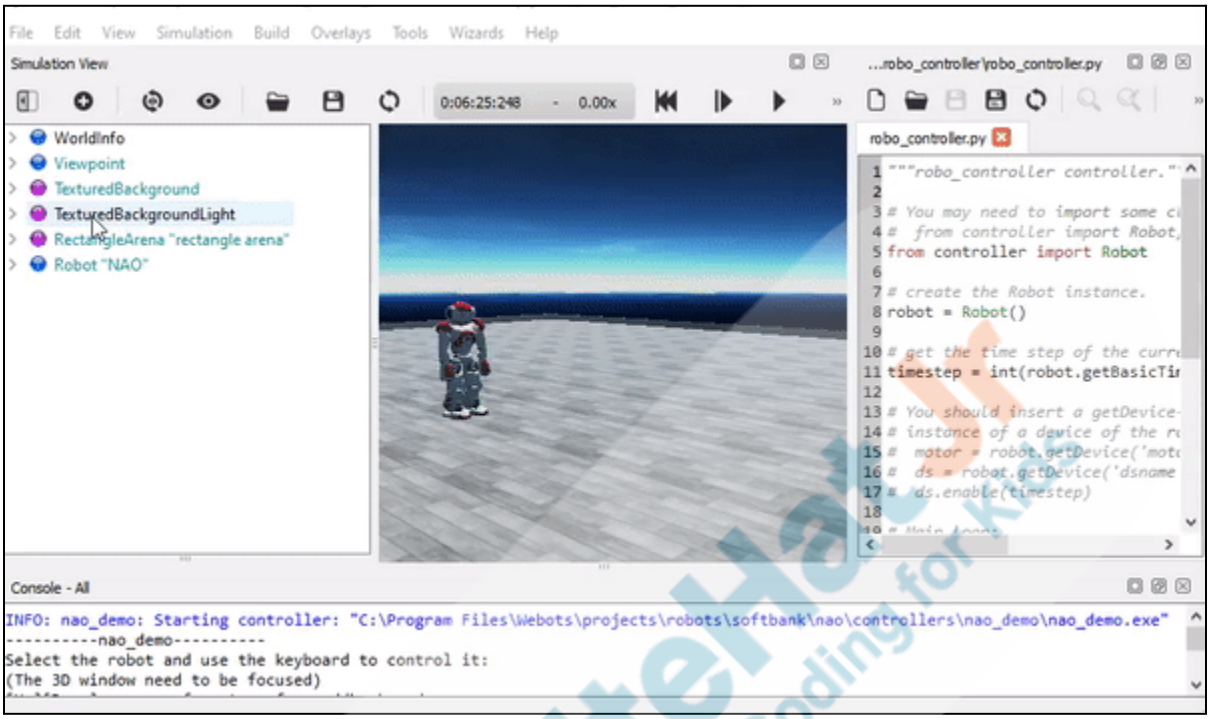
© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



Click here to view the new controller.

3. Now, assign this new controller to the **Nao** robot.

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



[Click here](#) to view the reference video.

Now, let's start writing the code. Delete the default code from the controller.

1. What should be the first line?

ESR: We need to import the classes which we need.

Correct!

Which classes do you think we will need?

ESR: We need the **Robot** class.

Great! And we will import the **Keyboard** class, which will help us control the robot with the **Keyboard.**

We will also import the **Motion** class to work with motion files.

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

```
from controller import Robot, Keyboard, Motion
```

2. Initiate **timestep** variable to **32** and create an instance of the Robot class.

```
timestep=32

robot= Robot()
```

3. Create an instance of **Keyboard** class and enable the keyboard.

```
keyboard = Keyboard()
keyboard.enable(timestep)

```

4. Let's initiate variables which will hold the motion files.

```
forward = Motion('../../motions/Forwards50.motion')
backward = Motion('../../motions/Backwards.motion')
sideStepLeft = Motion('../../motions/SideStepLeft.motion')
sideStepRight = Motion('../../motions/SideStepRight.motion')

```

5. Define **printMessages()** which will print some instructions for the user.

```
def printMessages():
    print('press up to move forward')
    print('press down to move backward')
    print('press left for side step left')
    print('press right for side step right')
```

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

6. Let's define a **startMotion()** method which will help us play our desired motion file.

```
def startMotion(motion):
    motion.play()
```

7. Initiate variable named **key** at **-1** and call the **printMessages()** method.

```
key = -1

printMessages()
```

8. Let's write the main loop now which will perform all the simulation steps.

```
while robot.step(timestep) != -1:
```

9. Now, we will use the **getKey()** method to get the key pressed by the user.

```
key = keyboard.getKey()
```

10. Write an **if-else** ladder to play different motions depending on the keys pressed.

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

```
if key == Keyboard.LEFT:
    startMotion(sideStepLeft)
elif key == Keyboard.RIGHT:
    startMotion(sideStepRight)
elif key == Keyboard.UP:
    startMotion(forward)
elif key == Keyboard.DOWN:
    startMotion(backward)
```

Reference Code:

```
from controller import Robot, Keyboard, Motion

timestep=32

robot= Robot()
keyboard = Keyboard()
keyboard.enable(timestep)


forward = Motion('../../motions/Forwards50.motion')
backward = Motion('../../motions/Backwards.motion')
sideStepLeft = Motion('../../motions/SideStepLeft.motion')
sideStepRight = Motion('../../motions/SideStepRight.motion')

def startMotion(motion):
    motion.play()

def printMessages():
    print('press up to move forward')
    print('press down to move backward')
    print('press left for side step left')
    print('press right for side step right')

key = -1

printMessages()
```

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

```
while robot.step(timestep) != -1:
    key = keyboard.getKey()

    if key == Keyboard.LEFT:
        startMotion(sideStepLeft)
    elif key == Keyboard.RIGHT:
        startMotion(sideStepRight)
    elif key == Keyboard.UP:
        startMotion(forward)
    elif key == Keyboard.DOWN:
        startMotion(backward)
```

Reference Output:

Click here to view the output video.

Great work!

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

Teacher Guides Student to Stop Screen Share

WRAP-UP SESSION - 05 mins

Activity details

Following are the WRAP-UP session deliverables:

- Appreciate the student.
- Revise the current class activities.
- Discuss the quizzes.

WRAP-UP QUIZ
Click on In-Class Quiz

Activity Details

Following are the session deliverables:

- Explain the facts and trivia
- Next class challenge
- Project for the day
- Additional Activity (Optional)

FEEDBACK

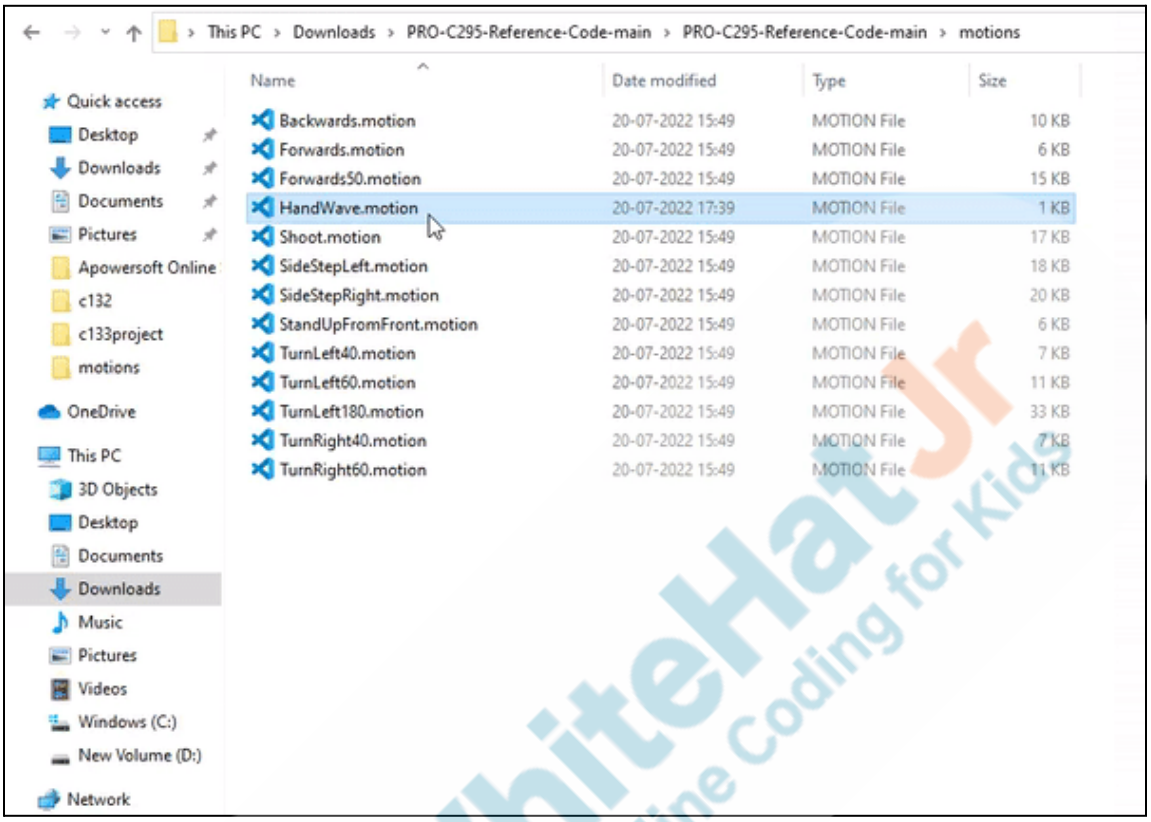
- **Appreciate and compliment the student for trying to learn a difficult concept.**
- **Get to know how they are feeling after the session.**
- **Review and check their understanding.**

Teacher Action	Student Action
You get “hats-off” for your excellent work! In the next class, we will learn about a floor cleaning robot.	*Make sure you have given at least 2 hats-off during the class for:*  

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

	
PROJECT OVERVIEW DISCUSSION Refer the document below in Activity Links Sections	
Teacher Clicks 	
ADDITIONAL ACTIVITIES (Optional)	
Teacher Action	**Student Action**
Let's create a simple salute motion using a motion file. 1. Create a copy of the HandWave.motion file in the motion folder. 2. Rename it as **salute.motion**	

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



Click here to view the reference code.

3. Open this file with visual studio code and delete everything after the second line.	
4. Now, first we need to raise the robot's right hand. We will change the value of **RShoulderPitch**. Because this is the joint which will help us move the hand up. Click here to view the reference video. Let's put the **RShoulderPitch** as **-1.2**	

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

```
#WEBOTS_MOTION,V1.0,RShoulderPitch,RShoulderRoll,LShoulderPitch
00:00:000,Pose1,-1.2,
```

5. Now, we need to move the hand a little away from its body. If you notice, there is another motor in the same joint.

   This is called a **Hinge2Joint.** A **Hinge2Joint** can move in 2 planes.

   To move the hand a little away from the body, we will use the second rotational motor in the same joint i. e. **RShoulderRoll**

   Assign **RShoulderRoll** to **-0.2**

6. Now, we need to bend the robot's elbow towards its head. Let's find the motor name operating this joint by clicking on the joint.

   Click here to view the reference video.

   Assign **RElbowRoll** to **1.5** in the **salute.motion** file.

```
1  #WEBOTS_MOTION,V1.0,RShoulderPitch,RShoulderRoll,RElbowRoll
2  00:00:000,Pose1,-1.2,-0.2,1.5
3
```

We need to change the code in the controller so that it can read the **salute.motion** file.

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

robo_controller.py*

```
1 from controller import Robot, Keyboard, Motion
2
3 timestep=32
4
5 robot= Robot()
6 keyboard = Keyboard()
7 keyboard.enable(timestep)
8
9 forward = Motion('../../motions/Forwards50.motion')
10 backward = Motion('../../motions/Backwards.motion')
11 sideStepLeft = Motion('../../motions/SideStepLeft.motion')
12 sideStepRight = Motion('../../motions/SideStepRight.motion')
13 salute = Motion('../../motions/salute.motion')
14
15 def startMotion(motion):
16     motion.play()
17
18 def printMessages():
19     print('press up to move forward')
20     print('press down to move backward')
21     print('press left for side step left')
22     print('press right for side step right')
23     print('press S to salute')
24
25 key = -1
26
```

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

robo_controller.py

```
    print('press up to move forward')
    print('press down to move backward')
    print('press left for side step left')
    print('press right for side step right')
    print('press S to salute')

key = -1

printMessages()

while robot.step(timestep) != -1:

    key = keyboard.getKey()
    print(key)

    if key == Keyboard.LEFT:
        startMotion(sideStepLeft)
    elif key == Keyboard.RIGHT:
        startMotion(sideStepRight)
    elif key == Keyboard.UP:
        startMotion(forward)
    elif key == Keyboard.DOWN:
        startMotion(backward)
    # 83 represents s key
    elif key == 83:
        startMotion(salute)
```

Now if you run this file and press S, we will get the following output.

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

Click here to view the reference video.

This is good but we would also want the first two fingers to open up. The motor for the right hand finger joints. The first two fingers are made up of joints named **RPhalanx1, RPhalanx2, RPhalanx3, RPhalanx4, RPhalanx5, RPhalanx6.** The thumb is made up of motors named **RPhalanx7 and RPhalanx8**. We want to open the first two fingers. And close the thumb.	

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

The teacher shows by experimenting with different values for the finger joints to determine the following values..

So, assign the **RPhalanx1, RPhalanx2, RPhalanx3, RPhalanx4, RPhalanx5, RPhalanx6** values as **1** to open the first two fingers**.** For **RPhalanx7, RPhalanx8 to** keep the values as 0 to keep the thumb closed.

This completes the salute posture.

```
#WEBOTS_MOTION,V1.0,RShoulderPitch,RShoulderRoll,RElbowRoll,RPhalanx1,RPhalanx2,RPhalanx3,RPhalanx4,RPhalanx5,RPhalanx6,RPhalanx7,RPhalanx8
00:01:000,Pose1,-1.2,-0.2,1.5,1,1,1,1,1,1,0,0
```

After some time, we would like the robot to go back to its regular pose. For that we will add a second line for pose.

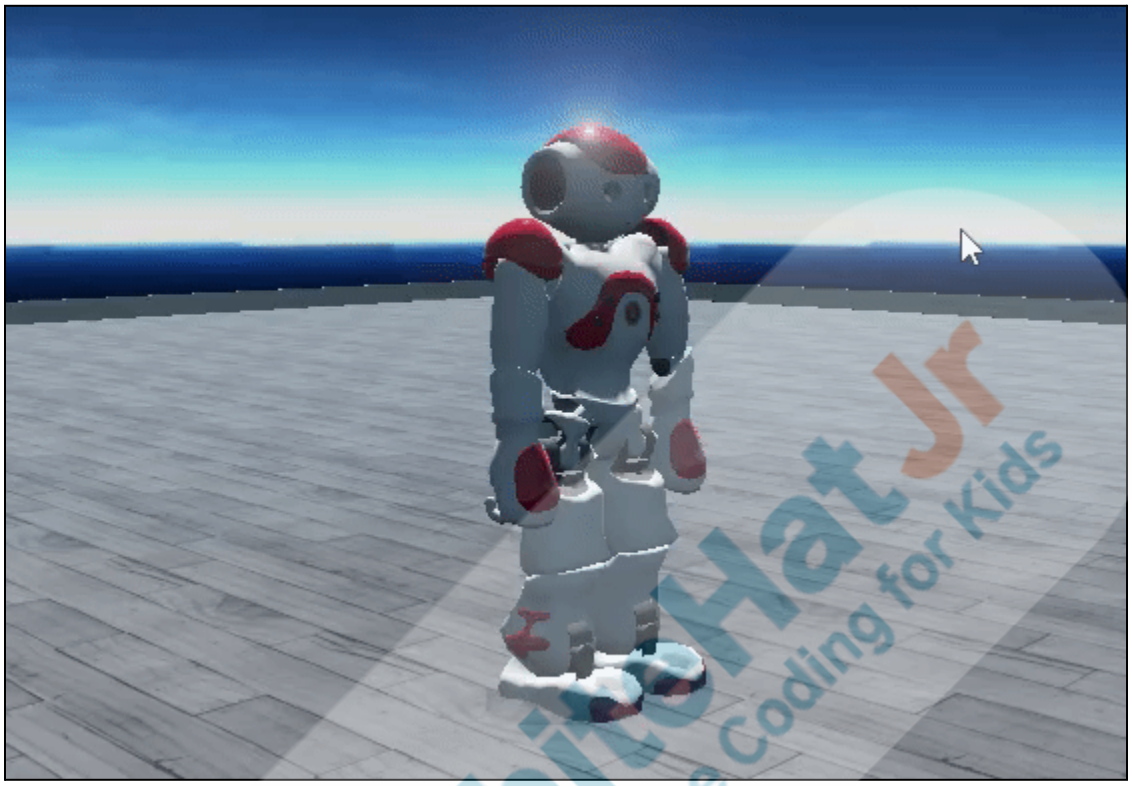
The pose will be executed for the first second and the second pose will be executed for the 2nd second.

```
#WEBOTS_MOTION,V1.0,RShoulderPitch,RShoulderRoll,RElbowRoll,RPhalanx1,RPhalanx2,RPhalanx3,RPhalanx4,RPhalanx5,RPhalanx6,RPhalanx7,RPhalanx8
00:01:000 Pose1,-1.2,-0.2,1.5,1,1,1,1,1,1,0,0
00:02:000,Pose2,1.5,0, 0,0,0,0,0,0,0,0,0
```

Click here to view the image.

Reference output:

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



Click here to view the reference output.

ACTIVITY LINKS		
Activity Name	**Description**	**Links**
Teacher Activity 1	Documentation for Motion functions	https://cyberbotics.com/doc/reference/motion
Teacher Reference 1	Reference Code	https://github.com/procodingclass/PRO-C295-Reference-Code
Teacher Reference 2	Motion files for Additional Activity	https://github.com/procodingclass/PRO-C295-Reference-Code-AA
Teacher Reference 3	In-Class Quiz	https://s3-whjr-curriculum-uploads.whjr.online/5cbf8e36-1f63-481c-96

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

		a7-63336657d543.pdf
Student Activity 1	Boilerplate Code	https://github.com/procodingclass/PRO-C295-Student-Boilerplate
Student Activity 2	Floor Texture	https://github.com/procodingclass/PRO-C295-Student-Boilerplate/blob/main/Wood-Tile.jpg
Student Activity 3	Documentation for Motion functions	https://cyberbotics.com/doc/reference/motion

© 2021 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.