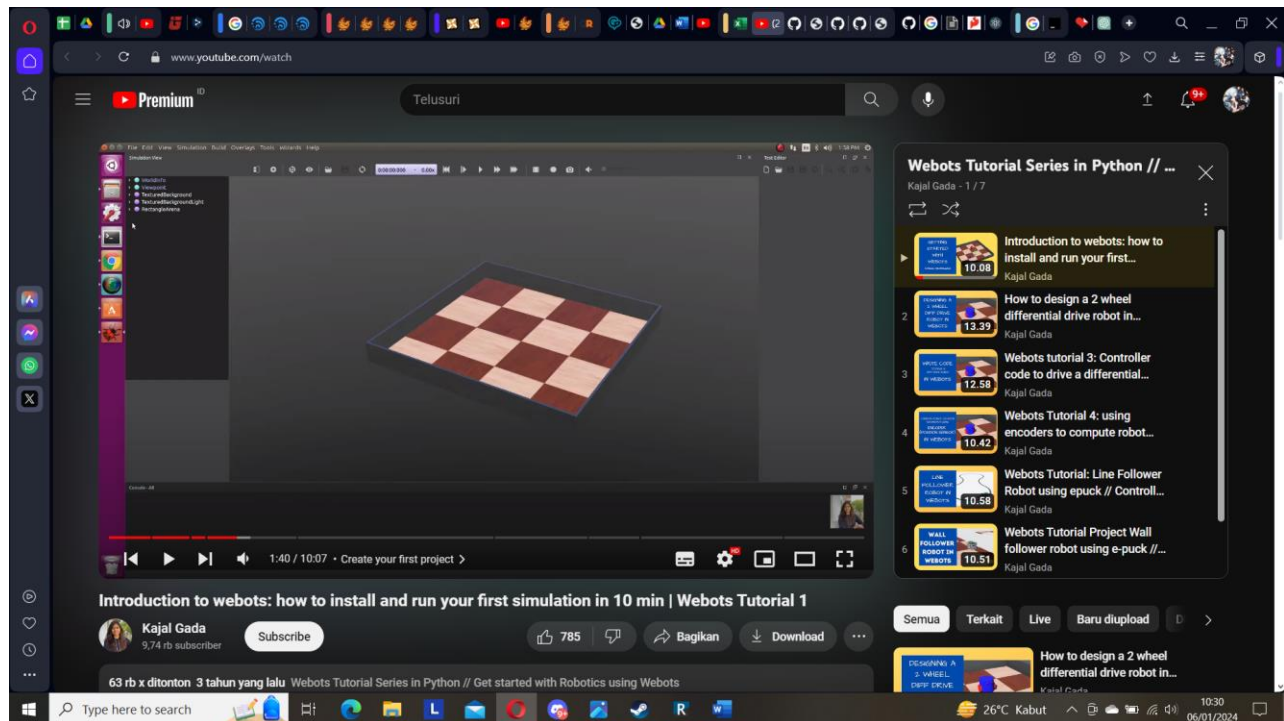


Nama : Rizqullah Imamuddin Habibi
NIM : 1103204139

Lecture 10

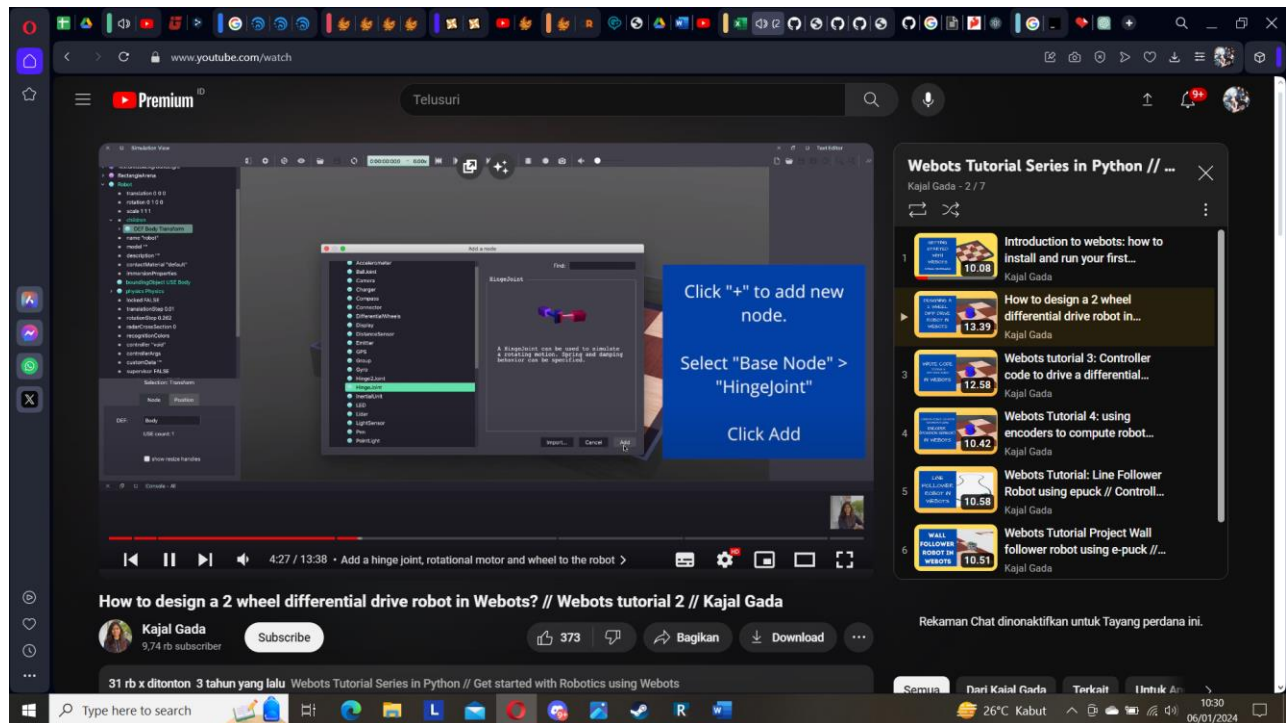


Dalam tutorial ini, Webots diperkenalkan sebagai perangkat lunak simulasi robot yang dirancang untuk membantu pengguna memahami konsep dasar robotika. Pembuat video memandu penonton melalui proses instalasi Webots pada sistem Ubuntu dan memberikan langkah-langkah awal untuk membuat proyek baru. Pembuat video menjelaskan cara menambahkan objek, seperti bola, ke dalam simulasi dan mengatur propertinya, termasuk aspek fisika.

Selanjutnya, pembuat video mendemonstrasikan pembuatan robot dengan pengontrol sendiri dalam Webots. Dia menggambarkan konsep pengkodean dasar menggunakan bahasa seperti C, C++, dan Python, memberikan penonton pemahaman tentang bagaimana mengedit dan mengelola perilaku robot dalam simulasi. Tutorial ini memberikan gambaran umum tentang fungsionalitas Webots, mulai dari menciptakan objek hingga membuat robot yang dapat diatur untuk berinteraksi dalam lingkungan simulasi.

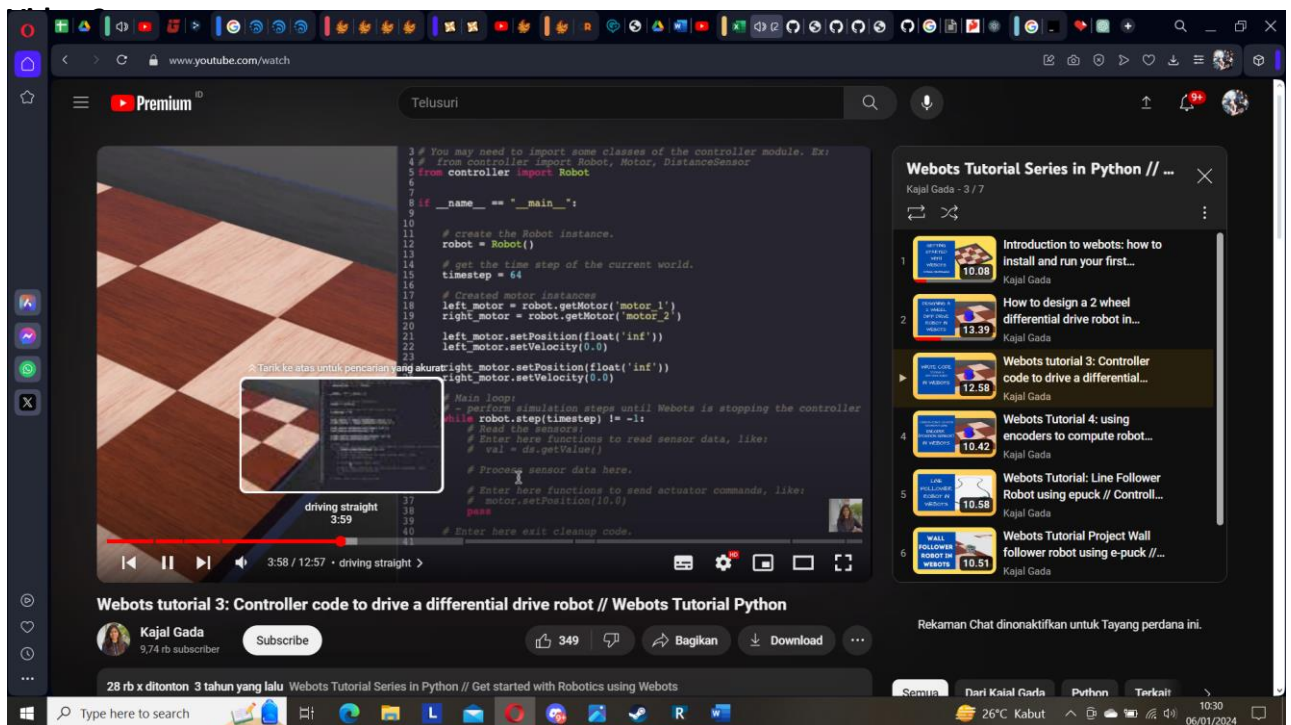
Pembuat video juga menyoroti kemampuan untuk menggunakan kembali objek dan robot yang sudah ada, serta memberikan wawasan singkat tentang pengaturan simulasi, menjelaskan cara menjalankan, menghentikan, dan mengatur ulang simulasi. Dengan memberikan pandangan menyeluruh tentang Webots, tutorial ini memberikan dasar bagi pemula untuk mulai menjelajahi dunia simulasi robotika menggunakan perangkat lunak tersebut.

Video 2



Dalam tutorial ini, Kajaal membimbing penonton dalam membuat robot penggerak diferensial dalam simulasi Webots menggunakan perangkat lunak vbots. Proses dimulai dengan membuat proyek baru, menambahkan arena persegi panjang, dan menciptakan struktur dasar robot. Kajaal mendemonstrasikan langkah-langkah untuk membuat badan robot, menambahkan roda dengan sambungan engsel, dan memberikan motor pada masing-masing roda untuk memungkinkan pergerakan. Selain itu, ia menyoroti penambahan elemen seperti mata dan memberikan detail tentang penyesuaian tampilan dan fisika untuk membuat robot terlihat realistis.

Kajaal juga memberikan instruksi tentang cara menggunakan kembali objek seperti bentuk dan fisika yang sudah dibuat sebelumnya. Tutorial ini menunjukkan penggunaan transformasi, sambungan engsel, motor, dan pengkodean untuk menciptakan robot yang dapat bergerak. Kajaal berjanji untuk mengajarkan cara menulis pengontrol agar robot dapat bergerak maju, berbelok, dan berputar di tempatnya dalam tutorial berikutnya. Ia juga memberikan referensi ke sumber daya tambahan dan dunia bot Wii sebagai inspirasi. Video ini ditutup dengan undangan untuk memberikan suka dan berlangganan jika penonton menikmati kontennya.



Dalam video tutorial ini, KAJAL kembali untuk memandu penonton melalui proses menulis kode pengontrol untuk menggerakkan robot penggerak diferensial dua roda dalam simulasi Webot menggunakan perangkat lunak vbots. Setelah membuat proyek dan struktur dasar robot, KAJAL mengajarkan cara mengemudikan robot lurus dan berputar di tempat. Dia menjelaskan konsep penghitungan waktu dan kecepatan untuk mengatur pergerakan robot dalam poligon, seperti persegi atau bentuk lainnya. KAJAL juga menggali ide sistem loop terbuka dan sistem loop tertutup untuk meningkatkan respons robot terhadap lingkungan sekitarnya.

Selanjutnya, KAJAL menunjukkan cara mengatur robot agar mengikuti pola gerakan berbentuk persegi, menghitung waktu dan kecepatan rotasi, serta menjelaskan bagaimana sistem loop tertutup dapat meningkatkan presisi robot. Video ini memberikan pemahaman yang baik tentang konsep dasar pengendalian robot menggunakan kode Python di lingkungan simulasi Webot. KAJAL mengajak penonton untuk berlangganan jika mereka menikmati seri tutorialnya dan menawarkan keterlibatan melalui komentar untuk pertanyaan atau klarifikasi.

Dengan pendekatan yang jelas dan ringan, KAJAL menyampaikan materi ini dengan baik, membantu penonton untuk memahami langkah-langkah teknis dan konsep dasar pengendalian robot dalam lingkungan simulasi.

Video 4

- Node Pelanggan (demo_topic_subscriber.cpp)

1. Inisialisasi:

```
ros::init(argc, argv,  
"demo_topic_subscriber");ros::NodeHandle  
node_obj;
```

Menginisialisasi node pelanggan dengan nama unik (demo_topic_subscriber) dan membuat objek NodeHandle.

2. Inisialisasi Pelanggan:

```
ros::Subscriber    number_subscriber    =    node_obj.subscribe("/numbers",    10,  
number_callback);
```

Membuat pelanggan untuk topik /numbers dengan ukuran buffer 10 dan menentukan fungsi panggilan balik (number_callback) yang akan dieksekusi saat data diterima.

3. Fungsi Panggilan Balik:

```
void number_callback(const std_msgs::Int32::ConstPtr& msg) {  
    ROS_INFO("Menerima [%d]", msg->data);  
}
```

Video 5

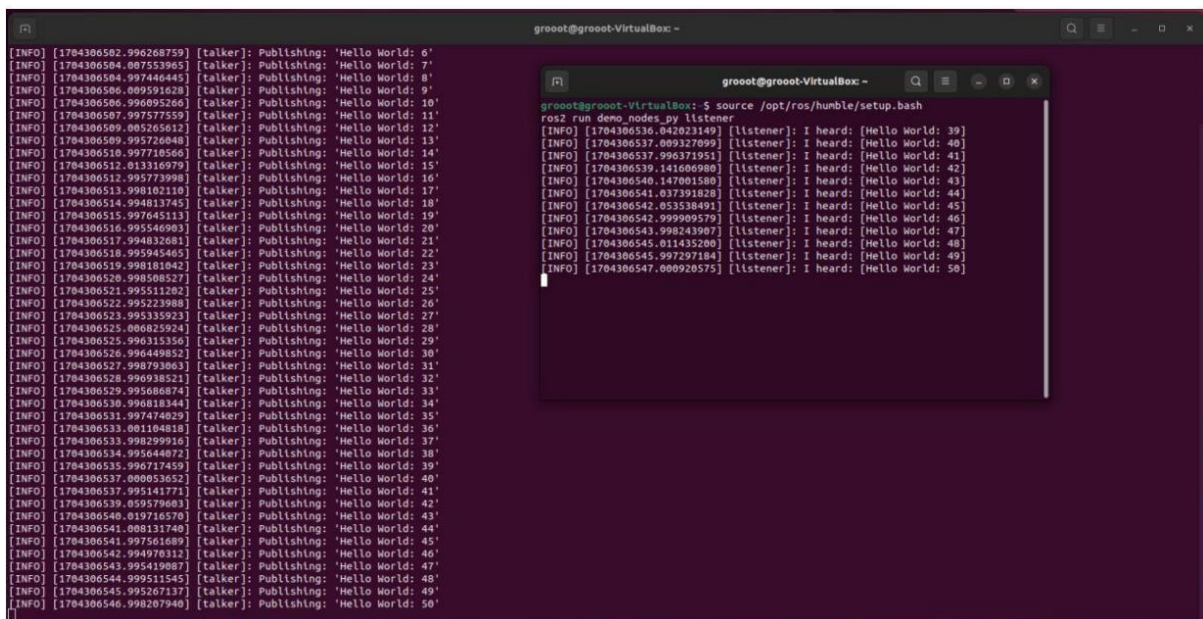
Mencetak nilai integer yang diterima saat panggilan balik dipicu.

- Spin:
ros::spi
n());

Menjaga node tetap aktif, menunggu pesan masuk, dan menjalankan fungsi panggilan balik.

File CMakeLists.txt dikonfigurasi untuk membangun kedua node (demo_topic_publisher dan demo_topic_subscriber). Pesan kustom diperkenalkan menggunakan file .msg (demo_msg.msg), dan kompilasinya dikelola dalam file CMakeLists.txt. Sebuah layanan kustom diperkenalkan menggunakan file .srv (demo_srv.srv), dan kompilasinya dikelola dengan cara yang sama. Artikel memberikan petunjuk langkah demi langkah tentang cara membangun, menjalankan, dan menguji node, serta perintah tambahan untuk debugging dan memahami komunikasi antar node.

Contoh:



```
groot@groot-VirtualBox: -  
[INFO] [1704306502.996268759] [talker]: Publishing: 'Hello World: 6'  
[INFO] [1704306504.007553965] [talker]: Publishing: 'Hello World: 7'  
[INFO] [1704306504.997446445] [talker]: Publishing: 'Hello World: 8'  
[INFO] [1704306506.009591628] [talker]: Publishing: 'Hello World: 9'  
[INFO] [1704306506.996095266] [talker]: Publishing: 'Hello World: 10'  
[INFO] [1704306507.997577559] [talker]: Publishing: 'Hello World: 11'  
[INFO] [1704306509.005265612] [talker]: Publishing: 'Hello World: 12'  
[INFO] [1704306509.995726048] [talker]: Publishing: 'Hello World: 13'  
[INFO] [1704306510.997716566] [talker]: Publishing: 'Hello World: 14'  
[INFO] [1704306512.013316979] [talker]: Publishing: 'Hello World: 15'  
[INFO] [1704306512.995773998] [talker]: Publishing: 'Hello World: 16'  
[INFO] [1704306513.998102110] [talker]: Publishing: 'Hello World: 17'  
[INFO] [1704306514.994813745] [talker]: Publishing: 'Hello World: 18'  
[INFO] [1704306515.997645113] [talker]: Publishing: 'Hello World: 19'  
[INFO] [1704306516.995546903] [talker]: Publishing: 'Hello World: 20'  
[INFO] [1704306517.994832681] [talker]: Publishing: 'Hello World: 21'  
[INFO] [1704306518.995945465] [talker]: Publishing: 'Hello World: 22'  
[INFO] [1704306519.998181042] [talker]: Publishing: 'Hello World: 23'  
[INFO] [1704306520.998508527] [talker]: Publishing: 'Hello World: 24'  
[INFO] [1704306521.995511202] [talker]: Publishing: 'Hello World: 25'  
[INFO] [1704306522.995223988] [talker]: Publishing: 'Hello World: 26'  
[INFO] [1704306523.995335923] [talker]: Publishing: 'Hello World: 27'  
[INFO] [1704306525.006025924] [talker]: Publishing: 'Hello World: 28'  
[INFO] [1704306525.996315356] [talker]: Publishing: 'Hello World: 29'  
[INFO] [1704306526.996449852] [talker]: Publishing: 'Hello World: 30'  
[INFO] [1704306527.998793063] [talker]: Publishing: 'Hello World: 31'  
[INFO] [1704306528.996938521] [talker]: Publishing: 'Hello World: 32'  
[INFO] [1704306529.995686874] [talker]: Publishing: 'Hello World: 33'  
[INFO] [1704306530.996818344] [talker]: Publishing: 'Hello World: 34'  
[INFO] [1704306531.997474029] [talker]: Publishing: 'Hello World: 35'  
[INFO] [1704306533.001104818] [talker]: Publishing: 'Hello World: 36'  
[INFO] [1704306533.998299916] [talker]: Publishing: 'Hello World: 37'  
[INFO] [1704306534.995644072] [talker]: Publishing: 'Hello World: 38'  
[INFO] [1704306535.996717459] [talker]: Publishing: 'Hello World: 39'  
[INFO] [1704306537.000853652] [talker]: Publishing: 'Hello World: 40'  
[INFO] [1704306537.995141771] [talker]: Publishing: 'Hello World: 41'  
[INFO] [1704306539.059379603] [talker]: Publishing: 'Hello World: 42'  
[INFO] [1704306540.019716570] [talker]: Publishing: 'Hello World: 43'  
[INFO] [1704306541.008131740] [talker]: Publishing: 'Hello World: 44'  
[INFO] [1704306541.997561689] [talker]: Publishing: 'Hello World: 45'  
[INFO] [1704306542.994978312] [talker]: Publishing: 'Hello World: 46'  
[INFO] [1704306543.995419087] [talker]: Publishing: 'Hello World: 47'  
[INFO] [1704306544.999511545] [talker]: Publishing: 'Hello World: 48'  
[INFO] [1704306545.995267137] [talker]: Publishing: 'Hello World: 49'  
[INFO] [1704306546.998207940] [talker]: Publishing: 'Hello World: 50'  
groot@groot-VirtualBox: -  
$ source /opt/ros/humble/setup.bash  
ros2 run demo_nodes_py listener  
[INFO] [1704306536.042023149] [listener]: I heard: [Hello World: 39]  
[INFO] [1704306537.009327099] [listener]: I heard: [Hello World: 40]  
[INFO] [1704306537.996371951] [listener]: I heard: [Hello World: 41]  
[INFO] [1704306539.141606980] [listener]: I heard: [Hello World: 42]  
[INFO] [1704306540.147001580] [listener]: I heard: [Hello World: 43]  
[INFO] [1704306541.037391828] [listener]: I heard: [Hello World: 44]  
[INFO] [1704306542.053538491] [listener]: I heard: [Hello World: 45]  
[INFO] [1704306542.999909579] [listener]: I heard: [Hello World: 46]  
[INFO] [1704306543.998243967] [listener]: I heard: [Hello World: 47]  
[INFO] [1704306545.011435200] [listener]: I heard: [Hello World: 48]  
[INFO] [1704306545.997297184] [listener]: I heard: [Hello World: 49]  
[INFO] [1704306547.000920575] [listener]: I heard: [Hello World: 50]
```