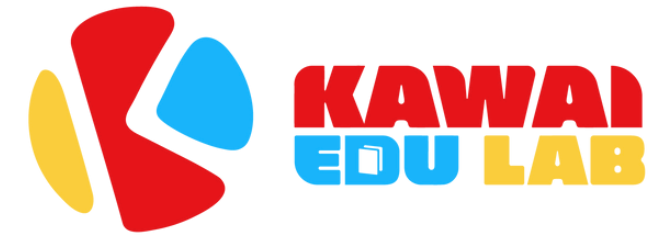
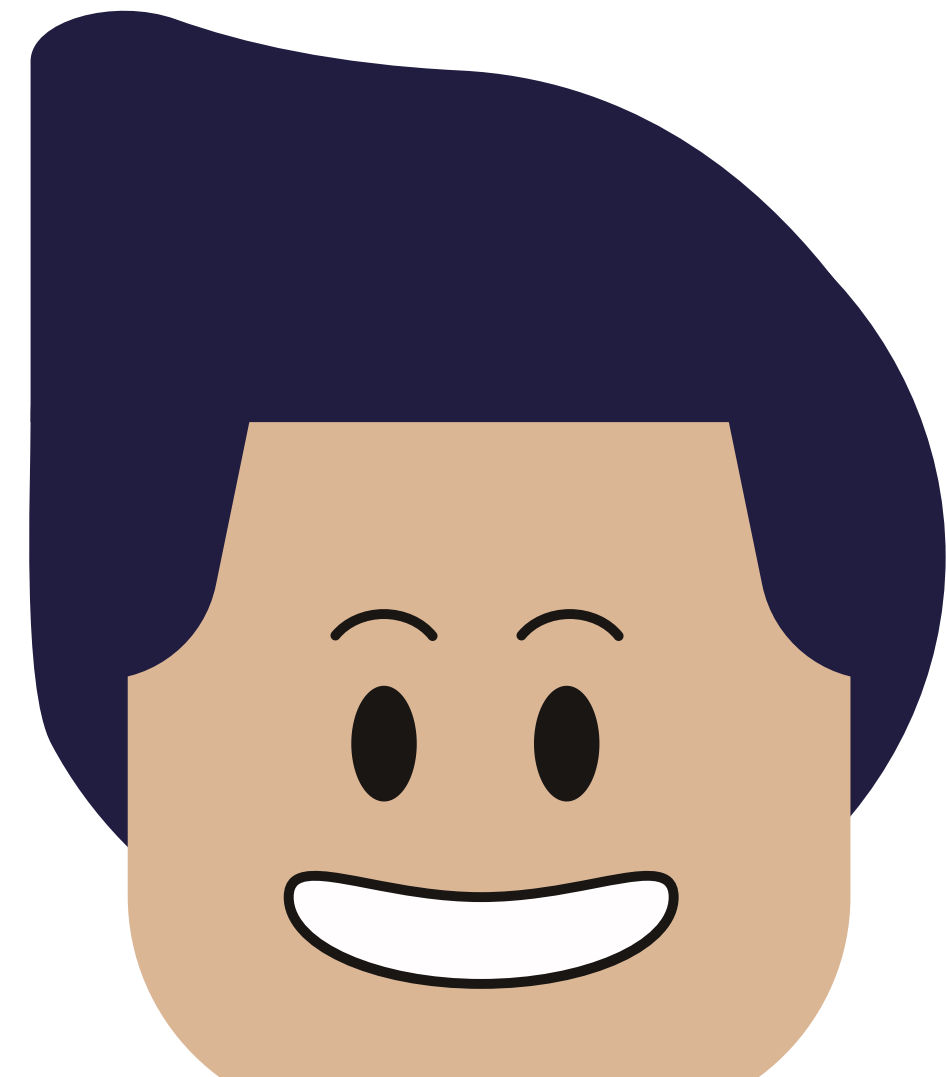
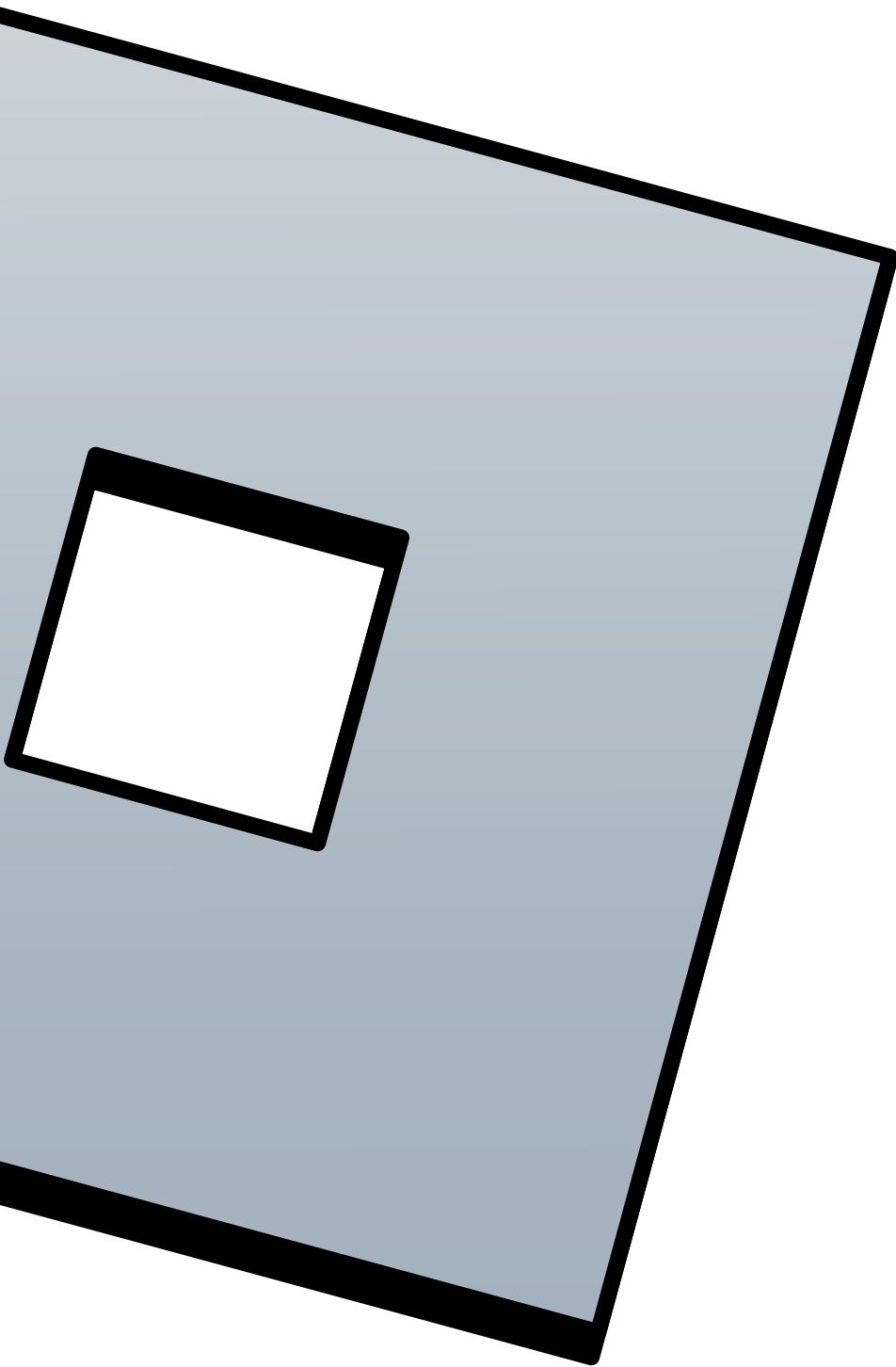


NEBULA



Pertemuan 10: Grouping and Components



Apa Itu Grouping dan Components?

Grouping dan Components adalah cara untuk mengelompokkan objek di dalam game dan membuatnya lebih mudah dikelola.

- ✓ Grouping: Menggabungkan beberapa objek menjadi satu kesatuan.
- ✓ Components: Objek atau script yang bisa digunakan kembali di berbagai tempat.

Dengan konsep ini, kita bisa membuat game lebih rapih, terorganisir, dan efisien!

Mengapa Menggunakan Grouping dan Components?

- Memudahkan pengelolaan → Objek yang berhubungan bisa disatukan dalam satu grup.
- Meningkatkan efisiensi → Components bisa digunakan berulang kali tanpa harus membuat ulang dari awal.
- Mempermudah scripting → Script bisa diterapkan ke seluruh grup sekaligus.

Misalnya, kita bisa membuat grup rumah yang terdiri dari tembok, atap, dan pintu. Atau kita bisa menggunakan komponen lampu yang bisa dipasang di berbagai tempat tanpa perlu coding ulang!

NEBULA



Mini Project: Box Animasi

Kita akan membuat kotak yang berputar, lalu mengubahnya menjadi komponen yang bisa digunakan di mana saja!

Langkah 1:

Membuat Objek dan Grouping

1. Buka Roblox Studio dan buat proyek baru atau gunakan "Baseplate".
2. Tambahkan beberapa Part
3. Buat kotak utama → Klik Home → Part → Block
4. Tambahkan detail seperti atap atau warna agar lebih menarik
5. Group Objek, Pilih semua Part. Klik Model → Group atau tekan Ctrl + G
6. Ganti nama grup di Explorer menjadi "RotatingBox"

Langkah 2: Membuat Primary Part

1. Atur PrimaryPart, Buka Properties (View → Properties).
2. Cari bagian PrimaryPart, Pilih salah satu part dalam grup (misalnya bagian dasar kotak).
3. Klik Set as PrimaryPart.

Dalam RotatingBox, kita menggunakan PrimaryPart sebagai titik pusat agar seluruh kotak bisa berputar dengan rapi tanpa terpecah! 🚀

Langkah 3:

Menulis Script Sederhana

Menambahkan Script untuk Animasi berputar (rotate)

```
1  local model = script.Parent
2  local primary = model.PrimaryPart -- Pastikan ada PrimaryPart yang ditetapkan
3
4  while true do
5      if primary then
6          model:SetPrimaryPartCFrame(primary.CFrame * CFrame.Angles(0, math.rad(5), 0))
7      end
8      wait(0.1)
9  end
10
```

Langkah 3: Membuatnya Jadi Komponen!

Sekarang kita bisa menggandakan dan menggunakan kembali objek ini dengan mudah!

- Klik RotatingBox, lalu Duplicate (Ctrl + D).
- Pindahkan ke lokasi lain, dan script-nya tetap akan berfungsi!
- Bisa digunakan di berbagai bagian game tanpa harus menulis ulang script!

NEBULA



Penjelasan

Bagaimana script ini bekerja? Yuk, kita bahas satu per satu!

`local model = script.Parent`

`local primary = model.PrimaryPart`

👉 Apa maksudnya?

- Menyimpan model (grup) dan mengambil PrimaryPart sebagai pusat rotasi

Penjelasan

while true do

🎮 Apa yang dilakukan bagian ini?

- Memulai perulangan tanpa henti, agar objek terus berputar.

if primary then

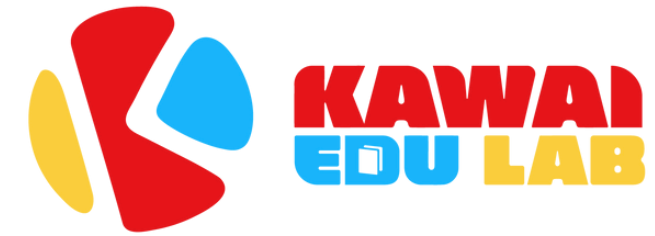
```
model:SetPrimaryPartCFrame(primary.CFrame * CFrame.Angles(0, math.rad(5), 0))
```

end

🎮 Apa yang dilakukan bagian ini?

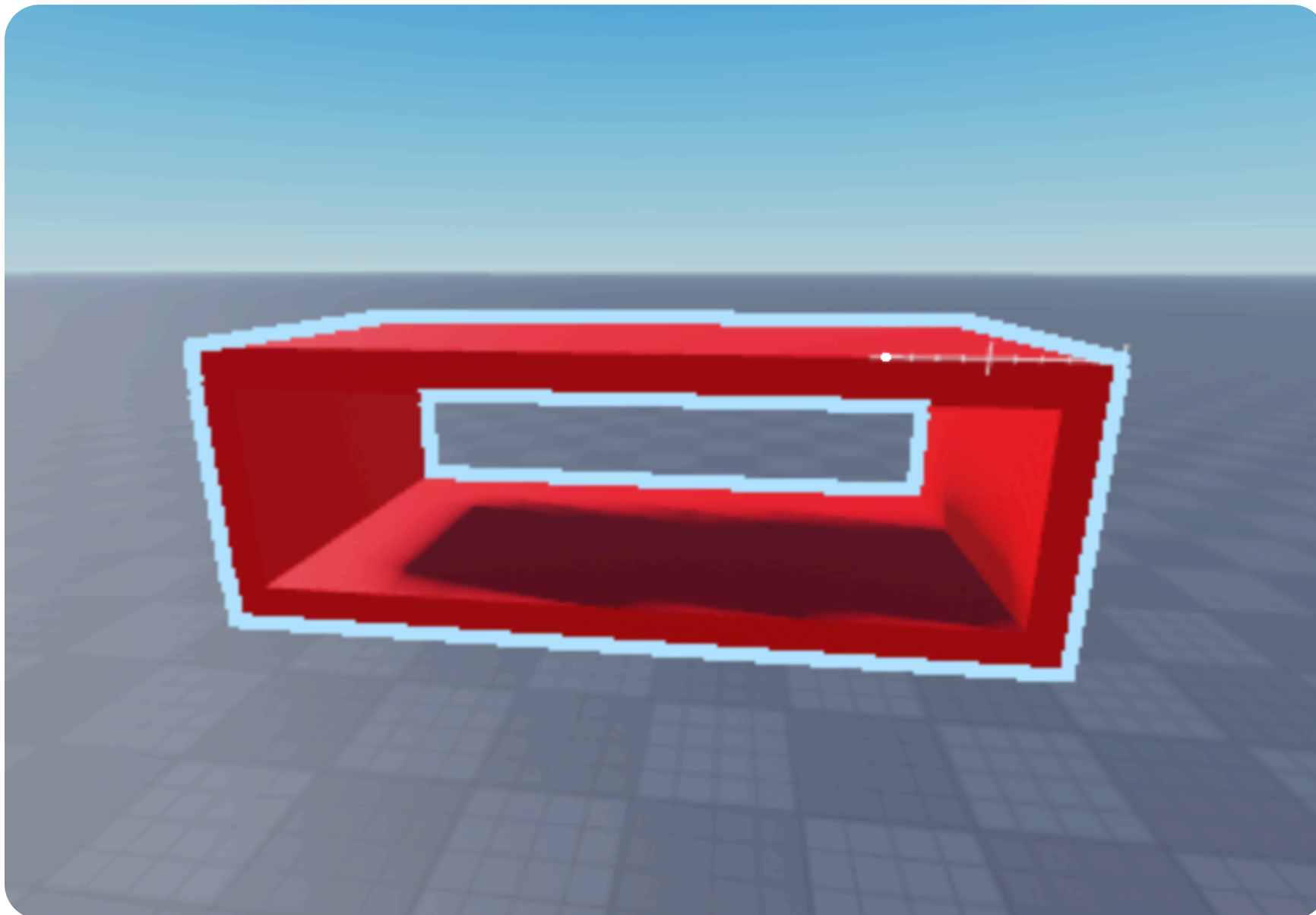
- Jika PrimaryPart ada, maka seluruh model diputar sebesar 5 derajat setiap iterasi.

NEBULA



**TEKAN
TOMBOL PLAY!**

Maka Model RotatingBox yang sudah kamu group tadi akan berputar sesuai dengan scriptnya!



Waktunya Berkreasi!

Sekarang saatnya kamu mencoba bereksperimen sendiri! 🔥

💡 Coba modifikasi proyek ini dengan ide berikut:

- Ubah kecepatan rotasi agar lebih cepat atau lebih lambat!
- Buat lebih dari satu kotak berputar di tempat yang berbeda!
- Coba terapkan konsep ini untuk objek lain, seperti pintu atau kipas angin!

Quiz!



Apa manfaat utama dari grouping di Roblox Studio?

- a. Membuat objek lebih ringan dalam game
- b. Memudahkan pengelolaan beberapa objek menjadi satu kesatuan**
- c. Menghapus objek secara otomatis

Bagaimana cara menggunakan komponen berulang kali di game?

- a. Menyalin objek dan menghapus script-nya
- b. Menggunakan script yang berbeda setiap objek
- c. Menggunakan objek yang sudah memiliki script dan menggandakannya**

NEBULA



**Jangan Lupa Save as
Hasil Karyamu ya!**