

- a. Apa saja yang harus dilakukan saat ingin merender suatu bentuk kubus?
- b. Jika ingin merubah bentuk kubus menjadi bentuk limas segi empat, apa saja yang harus dilakukan?

Jawab

- a. Untuk merender kubus di html dengan WebGL, yang harus dilakukan adalah:
  - 1. Membuat sebuah element canvas di HTML, kemudian menyimpannya sebagai sebuah variabel, misal “canvas” di Javascript (JS)
  - 2. Proses selanjutnya, akan dilakukan di JS
  - 3. Mengambil WebGLRenderingContext dan menyimpannya ke sebuah variabel, misal “gl” berdasarkan canvas pada no. 1 dengan cara “canvas.getContext(“webgl”)
  - 4. Menyiapkan vertex shader dan fragment shader, dapat ditulis di tag script pada HTML, ditulis dalam bentuk string dan disimpan pada sebuah variabel di JS, atau ditulis pada file dengan ekstensi .GLSL
  - 5. Membuat shader berdasarkan vertex shader dan fragment shader yang telah disiapkan
  - 6. Membuat program berdasarkan shader yang telah dibuat
  - 7. Menyiapkan array posisi/koordinat dari kubus yang akan dibuat
  - 8. Setiap sisi dari kubus merupakan gabungan dari 2 segitiga, sehingga kubus yang dibuat akan terdiri dari 12 segitiga dengan tiap segitiga terdiri dari 3 titik
  - 9. Jumlah total titik adalah 36 titik
  - 10. Menyiapkan warna untuk tiap titik, dapat dilakukan secara random, dan menyimpannya pada sebuah variabel
  - 11. Membuat buffer dan me-bind buffer tersebut ke ARRAY\_BUFFER untuk buffer posisi/koordinat dan warna kubus
  - 12. Mengisi/menyalin data ke buffer posisi/koordinat dan warna sesuai dengan data posisi/koordinat dan warna yang telah disiapkan
  - 13. Menyimpan lokasi atribut pada vertex shader yang diasosiasikan kepada posisi/koordinat dari kubus ke sebuah variabel
  - 14. Menyimpan lokasi atribut pada vertex shader yang diasosiasikan kepada warna dari kubus ke sebuah variabel
  - 15. Meng-enable atribut dan me-bind data ke atribut tersebut untuk atribut pada no 13 dan 14 dengan buffer posisi/koordinat dan warna pada no 12
  - 16. Mendefinisikan bagaimana tiap atribut mengambil data dari buffer-nya masing-masing dengan menggunakan vertexAttribPointer dengan size = 3, sebab setiap iterasi akan diambil 3 data (segitiga)
  - 17. Gunakan program yang telah dibuat pada no 6
  - 18. Meng-enable DEPTH\_TEST agar WebGL tidak merender titik/vertices dengan posisi yang lebih kecil, di belakang, dari titik/vertices lain pada sumbu-z, hal ini dilakukan agar kubus yang dibuat terlihat logis

19. Melakukan resize lebar dan tinggi canvas dengan menge-set-nya sesuai dengan dan lebar dan tinggi pada style di css, hal ini dilakukan sebab canvas memiliki 2 ukuran, yaitu drawingbuffer yang merupakan banyaknya piksel di canvas dan ukuran display yang ditentukan oleh css
  20. Resize dilakukan agar bentuk yang ditampilkan, nantinya, tidak terjadi stretch
  21. Menge-set ukuran clip space dari (0,0) hingga (lebar, tinggi) dengan viewport
  22. Membersihkan COLOR\_BUFFER\_BIT dan DEPTH\_BUFFER\_BIT
  23. Memanggil fungsi drawArrays primitiveType = gl.TRIANGLE, offset = 0, dan count = 36
- b. Untuk mengubah kubus menjadi limas segi empat
1. Mengubah data posisi/koordinat kubus agar sesuai menjadi sebuah limas segi empat
  2. Dengan demikian, jumlah titik/vertices akan menjadi 30, sebab limas segi empat memiliki 5 sisi
  3. Mengubah count pada drawArrays menjadi 30