

```
1 using UnityEngine;
2 using UnityEngine.SceneManagement;
3
4 public class Buttons : MonoBehaviour {
5
6     //LoadGame er kaldet når der trykkes på play knappen
7     public void LoadGame() {
8         //Loader game scenen som har tallet '1'
9         SceneManager.LoadSceneAsync(1);
10    }
11 }
12
```

```
1 //Lavet af Markus Brun Olsen uden for projektet
2
3 using UnityEngine;
4
5 public class CameraCrop : MonoBehaviour {
6     //fields
7     public Vector2 targetRatio = new Vector2(16, 9); // Set this to your target aspect ratio, eg. (16, 9) or (4, 3).
8     private Vector2 lastSize; //for remembering what the screen size was last frame
9
10    void Start() {
11        Camera.main.aspect = targetRatio.x / targetRatio.y; //setting screen aspect ratio (does not change it is just in case)
12        lastSize = new Vector2(-1, -1); //setting last size to something it can never be
13        FixedUpdate(); //setting crop of screen from the start
14    }
15
16    // Call this method if your window size or target aspect change.
17    public void FixedUpdate() {
18        if(lastSize.x != Screen.width || lastSize.y != Screen.height) { // screen size has changed
19            lastSize = new Vector2(Screen.width, Screen.height); //setting last screen size to current screen size (updating it)
20
21            //getting scaler
22            float widthScale = Screen.width / targetRatio.x;
23            float heightScale = Screen.height / targetRatio.y;
24
25            //choosing scaler
26            if(widthScale > heightScale) { //if adding black bars at width
27                float scale = (1f / widthScale) * heightScale; //getting scale
28                Camera.main.rect = new Rect((1f - scale) / 2f, 0f, scale, 1f); //setting black bars
29            } else if(widthScale < heightScale){
30                float scale = (1f / heightScale) * widthScale; //getting scale
31                Camera.main.rect = new Rect(0f, (1f - scale) / 2f, 1f, scale); //setting black bars
32            }
33        }
34    }
35 }
36
```

```
1 using UnityEngine;
2
3 public class Controller : MonoBehaviour {
4     //variabler
5     private Rigidbody rb;
6     public Camera cam;
7
8     //camera rotation
9     public float xRotationSpeed, yRotationSpeed;
10    private float xRotation, yRotation;
11
12    //bevægelse
13    public float acceleration, jumpSpeed;
14
15    // Maximal stråle distance for håndtags-tjek
16    public float maxRayDistance;
17
18    // Start er kaldet før første update frame
19    void Start() {
20        //sætter værdier
21        rb = GetComponent<Rigidbody>();
22
23        xRotation = 0.0f;
24        yRotation = 0.0f;
25    }
26
27    // Update kaldes en gang pr. frame
28    void Update() {
29        //bevægelse
30        //setter hvilken retning den bevæger sig i
31        Vector3 playerDir = new Vector3();
32        if(Input.GetKey("w") || Input.GetKey("up"))    playerDir += new Vector3(0, 0, +1); // frem
33        if(Input.GetKey("s") || Input.GetKey("down")) playerDir += new Vector3(0, 0, -1); // tilbage
34        if(Input.GetKey("a") || Input.GetKey("left")) playerDir += new Vector3(-1, 0, 0); // venstre
35        if(Input.GetKey("d") || Input.GetKey("right")) playerDir += new Vector3(+1, 0, 0); // højre
36
37        Vector3 worldDir = playerDir.x * cam.transform.right + playerDir.z
38            * cam.transform.forward; //får retningen men i verdnen
39        worldDir.y = 0; //sikre at man ikke kan flyve opad
40        rb.velocity += worldDir.normalized * acceleration *
41            Time.deltaTime; //tilføj bevægelse
42
43        //camera rotationen
44        xRotation -= xRotationSpeed * Input.GetAxis("Mouse Y");
45        yRotation += yRotationSpeed * Input.GetAxis("Mouse X");
46
47        //sikre at man max kan kigge lige op og minimum lige ned
48        if(xRotation > 90) xRotation = 90; //up
49        if(xRotation < -90) xRotation = -90; //ned
```

```
48
49     cam.transform.eulerAngles = new Vector3(xRotation, yRotation); // sætter rotationen
50 }
51
52 void FixedUpdate() {
53     /*
54     * Følgende kode er inspireret af unitys scripting manual:
55     * https://docs.unity3d.com/ScriptReference/Physics.Raycast.html
56     */
57
58     // Lav variable 'hit'
59     RaycastHit hit;
60
61     // Tjek om spilleren kigger på et håndtag
62     if(Physics.Raycast(cam.transform.position,
63         cam.transform.TransformDirection(Vector3.forward), out hit,
64         maxRayDistance)) {
65         if(hit.transform.gameObject.CompareTag("Lever") &&
66             Input.GetMouseButtonDown(0)) { // Hvis det er et håndtag
67             spilleren kigger på og han klikker på venstre muse-knap
68             hit.transform.gameObject.GetComponent<Gates>().output = !
69                 hit.transform.gameObject.GetComponent<Gates>().output; //
70                 Ændre håndtagets output til det modsatte af hvad det var
71
72                 //Debug.Log
73
74             ("Click");
75         }
76         //Debug.DrawRay
77         (cam.transform.position, cam.transform.TransformDirection
78         (Vector3.forward), Color.red);
79         //Debug.Log(hit.transform.gameObject.name + " " +
80             hit.transform.gameObject.tag);
81     }
82 }
83 }
```

```
1 using UnityEngine;
2
3 public class Gates : MonoBehaviour {
4     // Gatetype Variabler
5     public enum GateType { And, Or, Not, Nand, Nor, Xor, Lever };
6     public GateType gateType;
7
8     [Header("Inputs")]
9     // Gates og Håndtags gameobjekter tages som input
10    public GameObject gInput1;
11    public GameObject gInput2;
12
13    [HideInInspector]
14    // Output variable
15    public bool output = false;
16
17    // Input som boolean
18    private bool input1, input2;
19
20    // Start bliver kaldt ved første frame (Billede)
21    void Start() {
22
23    }
24
25    // Update kaldes ved hver frame (Billede)
26    void Update() {
27
28        if(gInput1 != null && gInput2 != null) { // Hvis begge inputs er sat. ➤
29            input1 = gInput1.GetComponent<Gates>().output; input2 = ➤
30                gInput2.GetComponent<Gates>().output; // Sætter gInputs output ➤
31                som vores input
32        } else if(gateType == GateType.Not && gInput1 != null) { // Hvis ➤
33            det er en not skal der kun et input til
34            input1 = gInput1.GetComponent<Gates>().output; // Sætter ➤
35            gInput1's output som input
36        } else { // Hvis ikke at nogen inputs er sat, gør begge input ➤
37            falske.
38            input1 = input2 = false;
39        }
40
41        if(gateType != GateType.Lever) { // Hvis det er en Logic gate og ➤
42            ikke et håndtag, fortsæt.
43            switch(gateType) { // Tjek igennem alle gate-muligheder og sæt ➤
44                output i forhold til.
45                case GateType.And:
46                    output = (input1 && input2);
47                    break;
48                case GateType.Or:
49                    output = (input1 || input2);
50                    break;
51                case GateType.Not:
52                    output = !input1;
```

```
46         break;
47     case GateType.Xor:
48         output = (input1 ^ input2);
49         break;
50     case GateType.Nand:
51         output = !(input1 && input2);
52         break;
53     case GateType.Nor:
54         output = !(input1 || input2);
55         break;
56     default:
57         Debug.LogError("Gatetype er ikke valgt");
58         output = false;
59         break;
60     }
61 }
62 }
63 }
```

```
1
2 using UnityEngine;
3
4 public class OpenDoors : MonoBehaviour {
5
6     // Variabler
7     private Vector3 startPosL, startPosR;
8     private GameObject doorL;
9     private GameObject doorR;
10
11     public GameObject gInput;
12
13     public bool closed;
14     private bool last, change;
15
16     public float speed;
17
18     // Start kaldes ved første frame
19     void Start() {
20
21         //sætter variabler
22         last = closed = true;
23
24         //sætter game objects
25         doorL = transform.Find("DoorL").gameObject;
26         doorR = transform.Find("DoorR").gameObject;
27
28         startPosL = doorL.transform.position;
29         startPosR = doorR.transform.position;
30     }
31
32     // Update kaldes ved hver frame
33     void Update() {
34         if(gInput != null) { // Hvis gInput findes
35             closed = !gInput.gameObject.GetComponent<Gates>().output; //
36             Sæt input til det modsatte af gInputs output
37         } else { // Ellers sæt closed til true
38             closed = true;
39         }
40
41         //checker for om der er en ændring
42         if(closed != last) {
43             last = closed;
44             change = true;
45         }
46
47         //gør noget ved det
48         if(change) {
49             if(closed) { //lukker sig
50                 float lastMag = (doorL.transform.position -
51                     startPosL).magnitude;
```

```
52         doorR.transform.Translate(-speed * Time.deltaTime, 0, 0);
53
54         if((doorL.transform.position - startPosL).magnitude >= lastMag) {
55             change = false;
56             doorL.transform.position = startPosL;
57             doorR.transform.position = startPosR;
58         }
59
60     } else { //åbner sig
61
62         doorL.transform.Translate(-speed * Time.deltaTime, 0, 0);
63         doorR.transform.Translate(+speed * Time.deltaTime, 0, 0);
64
65         if((doorL.transform.position - startPosL).magnitude > 1.5f) {
66             {
67                 change = false;
68                 doorL.transform.position = startPosL -
69                 transform.right.normalized * 1.5f;
70                 doorR.transform.position = startPosR +
71                 transform.right.normalized * 1.5f;
72             }
73         }
74     }
```



```
1 using UnityEngine;
2 using UnityEngine.SceneManagement;
3
4 public class Pause : MonoBehaviour {
5     // Update er kaldet på en gang pr frame
6     void Update() {
7
8         //køre en gang når man trykker på escape knappen
9         if(Input.GetKeyDown("escape")) {
10             SceneManager.LoadSceneAsync(0);
11         }
12     }
13 }
14 }
15
```

```
1 using UnityEngine;
2
3 public class Wire : MonoBehaviour {
4     // Input fra gates eller håndtag
5     public GameObject gInput;
6
7     // Materialer til når ledningen er tændt eller slukket
8     public Material wireOff, wireOn;
9
10    private bool input;
11
12    // Start kaldes ved første frame
13    void Start() {
14
15    }
16
17    // Update kaldes ved hver frame
18    void Update() {
19        if(gInput != null) { // Hvis gInput findes
20            input = gInput.GetComponent<Gates>().output; // Sæt ↗
21                input til gInputs output
22        } else { // Ellers sæt input til false
23            input = false;
24        }
25
26        if(wireOn == null || wireOff == null) { // Hvis ingen af materialer ↗
27            er sat hvis en fejl besked
28            Debug.LogError("Materialer er ikke sat");
29        } else if(input) { // Hvis input er aktiv
30            this.GetComponent<Renderer>().material = wireOn; // ↗
31                Sæt materialet til det aktive materiale
32        } else { // Ellers hvis slukkede
33            this.GetComponent<Renderer>().material = wireOff; // ↗
34                Sæt det tilbage til det slukkede materiale
35        }
36    }
37 }
```