using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.XR.ARFoundation;

using UnityEngine.UI;

public class SpawnableManager : MonoBehaviour

{

[SerializeField]

ARRaycastManager m\_RaycastManager;

List<ARRaycastHit> m\_Hits = new List<ARRaycastHit>();

[SerializeField]

GameObject colorPicker;

[SerializeField]

GameObject spawnablePrefab;

// first chair \*Fix to chairItemOne

GameObject scrollMenu;

GameObject spawnedObject;

GameObject chosenObject;

//HUD buttons

Button colorButton;

Button menuButton;

Button deleteButton;

public Button ExitButton;

public Sprite EditImage;

public Sprite DoneImage;

//Scroll menu chair item buttons.

public Button ChairItemOne;

public Button ChairItemTwo;

public Button ChairItemThree;

public Button ChairItemFour;

public Button ChairItemFive;

public Button ChairItemSix;

public Button ChairItemseven;

public Button ChairItemeight;

public Button ChairItemnine;

public Button ChairItemten;

public Button ChairItemeleven;

public Button ChairItemtwelve;

public Button ChairItemthirteen;

public Button ChairItemfourten;

public Button ChairItemfifteen;

//Scroll menu couch item buttons.

public Button CouchItemOne;

public Button CouchItemTwo;

public Button CouchItemThree;

public Button CouchItemFour;

public Button CouchItemFive;

public Button CouchItemSix;

public Button CouchItemseven;

public Button CouchItemeight;

public Button CouchItemnine;

public Button CouchItemten;

public Button CouchItemeleven;

public Button CouchItemtwelve;

public Button CouchItemthirteen;

public Button CouchItemfourten;

public Button CouchItemfifteen;

//Scroll menu table item buttons.

public Button TableItemOne;

public Button TableItemTwo;

public Button TableItemThree;

public Button TableItemFour;

public Button TableItemFive;

public Button TableItemSix;

public Button TableItemseven;

public Button TableItemeight;

public Button TableItemnine;

public Button TableItemten;

public Button TableItemeleven;

public Button TableItemtwelve;

public Button TableItemthirteen;

public Button TableItemfourten;

public Button TableItemfifteen;

//Scroll menu bed item buttons.

public Button bedItemOne;

public Button bedItemTwo;

public Button bedItemThree;

public Button bedItemFour;

public Button bedItemFive;

public Button bedItemSix;

public Button bedItemseven;

public Button bedItemeight;

public Button bedItemnine;

public Button bedItemten;

public Button bedItemeleven;

public Button bedItemtwelve;

public Button bedItemthirteen;

public Button bedItemfourten;

public Button bedItemfifteen;

//Scroll menu devices item buttons.

public Button devicesItemOne;

public Button devicesItemTwo;

public Button devicesItemThree;

public Button devicesItemFour;

public Button devicesItemFive;

public Button devicesItemSix;

public Button devicesItemseven;

public Button devicesItemeight;

public Button devicesItemnine;

public Button devicesItemten;

public Button devicesItemeleven;

public Button devicesItemtwelve;

public Button devicesItemthirteen;

public Button devicesItemfourten;

public Button devicesItemfifteen;

//Scroll menu healthytools item buttons.

public Button healthytoolsItemOne;

public Button healthytoolsItemTwo;

public Button healthytoolsItemThree;

public Button healthytoolsItemFour;

public Button healthytoolsItemFive;

public Button healthytoolsItemSix;

public Button healthytoolsItemseven;

public Button healthytoolsItemeight;

public Button healthytoolsItemnine;

public Button healthytoolsItemten;

public Button healthytoolsItemeleven;

public Button healthytoolsItemtwelve;

public Button healthytoolsItemthirteen;

public Button healthytoolsItemfourten;

public Button healthytoolsItemfifteen;

public FlexibleColorPicker fcp;

public Material material;

//Prefabs for the chair models.

public GameObject spawnableChair; // second chair \*Fix to chairItemTwo

public GameObject chairItemThree;

public GameObject chairItemFour;

public GameObject chairItemFive;

public GameObject chairItemSix;

public GameObject chairItemSeven;

public GameObject chairItemeight;

public GameObject chairItemnine;

public GameObject chairItemten;

public GameObject chairItemeleven;

public GameObject chairItemtwelve;

public GameObject chairItemthirteen;

public GameObject chairItemfourten;

public GameObject chairItemfifteen;

//Prefabs for the couch models.

public GameObject couchItemOne;

public GameObject couchItemTwo;

public GameObject couchItemThree;

public GameObject couchItemFour;

public GameObject couchItemFive;

public GameObject couchItemSix;

public GameObject couchItemseven;

public GameObject couchItemeight;

public GameObject couchItemnine;

public GameObject couchItemten;

public GameObject couchItemeleven;

public GameObject couchItemtwelve;

public GameObject couchItemthirteen;

public GameObject couchItemfourten;

public GameObject couchItemfifteen;

//Prefabs for the table models.

public GameObject tableItemOne;

public GameObject tableItemTwo;

public GameObject tableItemThree;

public GameObject tableItemFour;

public GameObject tableItemFive;

public GameObject tableItemSix;

public GameObject tableItemseven;

public GameObject tableItemeight;

public GameObject tableItemnine;

public GameObject tableItemten;

public GameObject tableItemeleven;

public GameObject tableItemtwelve;

public GameObject tableItemthirteen;

public GameObject tableItemfourten;

public GameObject tableItemfifteen;

//Prefabs for the bed models.

public GameObject bedOne;

public GameObject bedTwo;

public GameObject bedThree;

public GameObject bedFour;

public GameObject bedFive;

public GameObject bedSix;

public GameObject bedseven;

public GameObject bedeight;

public GameObject bednine;

public GameObject bedten;

public GameObject bedeleven;

public GameObject bedtwelve;

public GameObject bedthirteen;

public GameObject bedfourten;

public GameObject bedfifteen;

//Prefabs for the devices models.

public GameObject devicesOne;

public GameObject devicesTwo;

public GameObject devicesThree;

public GameObject devicesFour;

public GameObject devicesFive;

public GameObject devicesSix;

public GameObject devicesseven;

public GameObject deviceseight;

public GameObject devicesnine;

public GameObject devicesten;

public GameObject deviceseleven;

public GameObject devicestwelve;

public GameObject devicesthirteen;

public GameObject devicesfourten;

public GameObject devicesfifteen;

//Prefabs for the healthytools models.

public GameObject healthytoolsOne;

public GameObject healthytoolsTwo;

public GameObject healthytoolsThree;

public GameObject healthytoolsFour;

public GameObject healthytoolsFive;

public GameObject healthytoolsSix;

public GameObject healthytoolsseven;

public GameObject healthytoolseight;

public GameObject healthytoolsnine;

public GameObject healthytoolsten;

public GameObject healthytoolseleven;

public GameObject healthytoolstwelve;

public GameObject healthytoolsthirteen;

public GameObject healthytoolsfourten;

public GameObject healthytoolsfifteen;

bool pickerShown = false;

bool menuShown = false;

bool objectChosen = false;

/\*bool isPressed = false;

bool left = false;

bool right = false;\*/

/\*public Button LeftRotate;

public Button RightRotate;\*/

// Start is called before the first frame update

void Start()

{

//Locating and assigning HUD

colorButton = GameObject.Find("ColorButton").GetComponent<Button>();

colorButton.onClick.AddListener(ColorControl);

colorButton.gameObject.SetActive(false);

colorPicker.SetActive(false);

menuButton = GameObject.Find("MenuButton").GetComponent<Button>();

menuButton.onClick.AddListener(MenuControl);

ExitButton.onClick.AddListener(MenuControl);

ExitButton.gameObject.SetActive(false);

//LeftRotate.gameObject.SetActive(false);

//RightRotate.gameObject.SetActive(false);

// LeftRotate.onClick.AddListener(delegate { TogglePressed(true, 1); });

//RightRotate.onClick.AddListener(delegate { TogglePressed(true, 2); });

deleteButton = GameObject.Find("DeleteButton").GetComponent<Button>();

deleteButton.onClick.AddListener(DeleteObject);

deleteButton.gameObject.SetActive(false);

//Getting scrollmenu ready and hidden in the start

scrollMenu = GameObject.Find("ScrollMenu");

scrollMenu.SetActive(false);

spawnedObject = null; //Nullllllllllllllllllllllllllllllllllll boiii

//Chair item button assignments

/\*ChairItemOne = GameObject.Find("ChairItemOne").GetComponent<Button>();

ChairItemTwo = GameObject.Find("ChairItemTwo").GetComponent<Button>();

ChairItemThree = GameObject.Find("ChairItemThree").GetComponent<Button>();

ChairItemFour = GameObject.Find("ChairItemFour").GetComponent<Button>();

ChairItemFive = GameObject.Find("ChairItemFive").GetComponent<Button>();

ChairItemSix = GameObject.Find("ChairItemSix").GetComponent<Button>();\*/

//Chair items ObjectControl number assignments

ChairItemOne.onClick.AddListener(delegate { ObjectControl(1); });

ChairItemTwo.onClick.AddListener(delegate { ObjectControl(2); });

ChairItemThree.onClick.AddListener(delegate { ObjectControl(3); });

ChairItemFour.onClick.AddListener(delegate { ObjectControl(4); });

ChairItemFive.onClick.AddListener(delegate { ObjectControl(5); });

ChairItemSix.onClick.AddListener(delegate { ObjectControl(6); });

ChairItemseven.onClick.AddListener(delegate { ObjectControl(7); });

ChairItemeight.onClick.AddListener(delegate { ObjectControl(8); });

ChairItemnine.onClick.AddListener(delegate { ObjectControl(9); });

ChairItemten.onClick.AddListener(delegate { ObjectControl(10); });

ChairItemeleven.onClick.AddListener(delegate { ObjectControl(11); });

ChairItemtwelve.onClick.AddListener(delegate { ObjectControl(12); });

ChairItemthirteen.onClick.AddListener(delegate { ObjectControl(13); });

ChairItemfourten.onClick.AddListener(delegate { ObjectControl(14); });

ChairItemfifteen.onClick.AddListener(delegate { ObjectControl(15); });

//Couch item button assignments

/\* CouchItemOne = GameObject.Find("CouchItemOne").GetComponent<Button>();

CouchItemTwo = GameObject.Find("CouchItemTwo").GetComponent<Button>();

CouchItemThree = GameObject.Find("CouchItemThree").GetComponent<Button>();

CouchItemFour = GameObject.Find("CouchItemFour").GetComponent<Button>();

CouchItemFive = GameObject.Find("CouchItemFive").GetComponent<Button>();

CouchItemSix = GameObject.Find("CouchItemSix").GetComponent<Button>(); \*/

//Couch items ObjectControl number assignments

CouchItemOne.onClick.AddListener(delegate { ObjectControl(16); });

CouchItemTwo.onClick.AddListener(delegate { ObjectControl(17); });

CouchItemThree.onClick.AddListener(delegate { ObjectControl(18); });

CouchItemFour.onClick.AddListener(delegate { ObjectControl(19); });

CouchItemFive.onClick.AddListener(delegate { ObjectControl(20); });

CouchItemSix.onClick.AddListener(delegate { ObjectControl(21); });

CouchItemseven.onClick.AddListener(delegate { ObjectControl(22); });

CouchItemeight.onClick.AddListener(delegate { ObjectControl(23); });

CouchItemnine.onClick.AddListener(delegate { ObjectControl(24); });

CouchItemten.onClick.AddListener(delegate { ObjectControl(25); });

CouchItemeleven.onClick.AddListener(delegate { ObjectControl(26); });

CouchItemtwelve.onClick.AddListener(delegate { ObjectControl(27); });

CouchItemthirteen.onClick.AddListener(delegate { ObjectControl(28); });

CouchItemfourten.onClick.AddListener(delegate { ObjectControl(29); });

CouchItemfifteen.onClick.AddListener(delegate { ObjectControl(30); });

//Table item button assignments

/\*TableItemOne = GameObject.Find("TableItemOne").GetComponent<Button>();

TableItemTwo = GameObject.Find("TableItemTwo").GetComponent<Button>();

TableItemThree = GameObject.Find("TableItemThree").GetComponent<Button>();

TableItemFour = GameObject.Find("TableItemFour").GetComponent<Button>();

TableItemFive = GameObject.Find("TableItemFive").GetComponent<Button>();

TableItemSix = GameObject.Find("TableItemSix").GetComponent<Button>();\*/

//Table items ObjectControl number assignments

TableItemOne.onClick.AddListener(delegate { ObjectControl(31); });

TableItemTwo.onClick.AddListener(delegate { ObjectControl(32); });

TableItemThree.onClick.AddListener(delegate { ObjectControl(33); });

TableItemFour.onClick.AddListener(delegate { ObjectControl(34); });

TableItemFive.onClick.AddListener(delegate { ObjectControl(35); });

TableItemSix.onClick.AddListener(delegate { ObjectControl(36); });

TableItemseven.onClick.AddListener(delegate { ObjectControl(37); });

TableItemeight.onClick.AddListener(delegate { ObjectControl(38); });

TableItemnine.onClick.AddListener(delegate { ObjectControl(39); });

TableItemten.onClick.AddListener(delegate { ObjectControl(40); });

TableItemeleven.onClick.AddListener(delegate { ObjectControl(41); });

TableItemtwelve.onClick.AddListener(delegate { ObjectControl(42); });

TableItemthirteen.onClick.AddListener(delegate { ObjectControl(43); });

TableItemfourten.onClick.AddListener(delegate { ObjectControl(44); });

TableItemfifteen.onClick.AddListener(delegate { ObjectControl(45); });

//Bed items ObjectControl number assignments

bedItemOne.onClick.AddListener(delegate { ObjectControl(46); });

bedItemTwo.onClick.AddListener(delegate { ObjectControl(47); });

bedItemThree.onClick.AddListener(delegate { ObjectControl(48); });

bedItemFour.onClick.AddListener(delegate { ObjectControl(49); });

bedItemFive.onClick.AddListener(delegate { ObjectControl(50); });

bedItemSix.onClick.AddListener(delegate { ObjectControl(51); });

bedItemseven.onClick.AddListener(delegate { ObjectControl(52); });

bedItemeight.onClick.AddListener(delegate { ObjectControl(53); });

bedItemnine.onClick.AddListener(delegate { ObjectControl(54); });

bedItemten.onClick.AddListener(delegate { ObjectControl(55); });

bedItemeleven.onClick.AddListener(delegate { ObjectControl(56); });

bedItemtwelve.onClick.AddListener(delegate { ObjectControl(57); });

bedItemthirteen.onClick.AddListener(delegate { ObjectControl(58); });

bedItemfourten.onClick.AddListener(delegate { ObjectControl(59); });

bedItemfifteen.onClick.AddListener(delegate { ObjectControl(60); });

//devices items ObjectControl number assignments

devicesItemOne.onClick.AddListener(delegate { ObjectControl(61); });

devicesItemTwo.onClick.AddListener(delegate { ObjectControl(62); });

devicesItemThree.onClick.AddListener(delegate { ObjectControl(63); });

devicesItemFour.onClick.AddListener(delegate { ObjectControl(64); });

devicesItemFive.onClick.AddListener(delegate { ObjectControl(65); });

devicesItemSix.onClick.AddListener(delegate { ObjectControl(66); });

devicesItemseven.onClick.AddListener(delegate { ObjectControl(67); });

devicesItemeight.onClick.AddListener(delegate { ObjectControl(68); });

devicesItemnine.onClick.AddListener(delegate { ObjectControl(69); });

devicesItemten.onClick.AddListener(delegate { ObjectControl(70); });

devicesItemeleven.onClick.AddListener(delegate { ObjectControl(71); });

devicesItemtwelve.onClick.AddListener(delegate { ObjectControl(72); });

devicesItemthirteen.onClick.AddListener(delegate { ObjectControl(73); });

devicesItemfourten.onClick.AddListener(delegate { ObjectControl(74); });

devicesItemfifteen.onClick.AddListener(delegate { ObjectControl(75); });

//healthytools items ObjectControl number assignments

healthytoolsItemOne.onClick.AddListener(delegate { ObjectControl(76); });

healthytoolsItemTwo.onClick.AddListener(delegate { ObjectControl(77); });

healthytoolsItemThree.onClick.AddListener(delegate { ObjectControl(78); });

healthytoolsItemFour.onClick.AddListener(delegate { ObjectControl(79); });

healthytoolsItemFive.onClick.AddListener(delegate { ObjectControl(80); });

healthytoolsItemSix.onClick.AddListener(delegate { ObjectControl(81); });

healthytoolsItemseven.onClick.AddListener(delegate { ObjectControl(82); });

healthytoolsItemeight.onClick.AddListener(delegate { ObjectControl(83); });

healthytoolsItemnine.onClick.AddListener(delegate { ObjectControl(84); });

healthytoolsItemten.onClick.AddListener(delegate { ObjectControl(85); });

healthytoolsItemeleven.onClick.AddListener(delegate { ObjectControl(86); });

healthytoolsItemtwelve.onClick.AddListener(delegate { ObjectControl(87); });

healthytoolsItemthirteen.onClick.AddListener(delegate { ObjectControl(88); });

healthytoolsItemfourten.onClick.AddListener(delegate { ObjectControl(89); });

healthytoolsItemfifteen.onClick.AddListener(delegate { ObjectControl(90); });

}

// Update is called once per frame

void Update()

{

//Update color picked material color.

material.color = fcp.color;

if (Input.touchCount == 0)

return;

if (menuShown == false && pickerShown == false)

{

if (m\_RaycastManager.Raycast(Input.GetTouch(0).position, m\_Hits))

{

if (chosenObject != null && objectChosen && !pickerShown)

{

if (Input.GetTouch(0).phase == TouchPhase.Began)

{

// if (spawnedObject != null)

// Destroy(spawnedObject);

SpawnPrefab(m\_Hits[0].pose.position);

colorButton.gameObject.SetActive(true);

deleteButton.gameObject.SetActive(true);

//RightRotate.gameObject.SetActive(true);

//LeftRotate.gameObject.SetActive(true);

}

else if (Input.GetTouch(0).phase == TouchPhase.Moved && spawnedObject != null)

{

spawnedObject.transform.position = m\_Hits[0].pose.position;

}

/\* if (Input.GetTouch(0).phase == TouchPhase.Ended)

{

return;

} \*/

}

}

}

/\* if (isPressed)

{

if (left)

{

spawnedObject.transform.Rotate(Vector3.forward \* 20f \* Time.deltaTime);

}

if (right)

{

spawnedObject.transform.Rotate(Vector3.forward \* -20f \* Time.deltaTime);

}

}\*/

}

//Spawning prefab with assigned object from ObjectControl

private void SpawnPrefab(Vector3 spawnPosition)

{

material = chosenObject.GetComponent<Renderer>().material;

spawnedObject = Instantiate(chosenObject, spawnPosition, new Quaternion(0, -190, -180, 1));

}

/\* private void ButtonPressed()

{

buttonPressed = true;

colorButton.gameObject.SetActive(true);

}\*/

private void ColorControl()

{

if (spawnedObject != null)

{

if (!pickerShown)

{

pickerShown = true;

Debug.Log("nyt pit�is aueta v�ri ikkuna");

colorPicker.SetActive(true);

deleteButton.gameObject.SetActive(false);

colorButton.GetComponent<Image>().sprite = DoneImage;

}

else if (pickerShown)

{

colorPicker.SetActive(false);

deleteButton.gameObject.SetActive(true);

colorButton.GetComponent<Image>().sprite = EditImage;

pickerShown = false;

}

}

}

/\*public void TogglePressed(bool value, int x)

{

isPressed = value;

if (x == 1)

{

left = value;

right = false;

}

if (x == 2)

{

right = value;

left = false;

}

}\*/

private void MenuControl()

{

//Debug.Log("Menunappipainettu");

if (!menuShown)

{

menuShown = true;

menuButton.gameObject.SetActive(false);

deleteButton.gameObject.SetActive(false);

ExitButton.gameObject.SetActive(true);

scrollMenu.SetActive(true);

//LeftRotate.gameObject.SetActive(false);

//RightRotate.gameObject.SetActive(false);

}

else if (menuShown)

{

menuShown = false;

scrollMenu.SetActive(false);

menuButton.gameObject.SetActive(true);

ExitButton.gameObject.SetActive(false);

if (spawnedObject != null)

deleteButton.gameObject.SetActive(true);

}

}

//ObjectControl controls the chosen model from the item menu.

private void ObjectControl(int objectNumber)

{

MenuControl();

//

//Destroy(spawnedObject);

int caseNumber = objectNumber;

switch (caseNumber)

{

//Cases 1-15 chairs.

case 1:

chosenObject = spawnablePrefab;

objectChosen = true;

break;

case 2:

chosenObject = spawnableChair;

objectChosen = true;

break;

case 3:

chosenObject = chairItemThree;

objectChosen = true;

break;

case 4:

chosenObject = chairItemFour;

objectChosen = true;

break;

case 5:

chosenObject = chairItemFive;

objectChosen = true;

break;

case 6:

chosenObject = chairItemSix;

objectChosen = true;

break;

case 7:

chosenObject = chairItemSeven;

objectChosen = true;

break;

case 8:

chosenObject = chairItemeight;

objectChosen = true;

break;

case 9:

chosenObject = chairItemnine;

objectChosen = true;

break;

case 10:

chosenObject = chairItemten;

objectChosen = true;

break;

case 11:

chosenObject = chairItemeleven;

objectChosen = true;

break;

case 12:

chosenObject = chairItemtwelve;

objectChosen = true;

break;

case 13:

chosenObject = chairItemthirteen;

objectChosen = true;

break;

case 14:

chosenObject = chairItemfifteen;

objectChosen = true;

break;

case 15:

chosenObject = chairItemfifteen;

objectChosen = true;

break;

//Cases 16-30 couches.

case 16:

chosenObject = couchItemOne;

objectChosen = true;

break;

case 17:

chosenObject = couchItemTwo;

objectChosen = true;

break;

case 18:

chosenObject = couchItemThree;

objectChosen = true;

break;

case 19:

chosenObject = couchItemFour;

objectChosen = true;

break;

case 20:

chosenObject = couchItemFive;

objectChosen = true;

break;

case 21:

chosenObject = couchItemSix;

objectChosen = true;

break;

case 22:

chosenObject = couchItemseven;

objectChosen = true;

break;

case 23:

chosenObject = couchItemeight;

objectChosen = true;

break;

case 24:

chosenObject = couchItemnine;

objectChosen = true;

break;

case 25:

chosenObject = couchItemten;

objectChosen = true;

break;

case 26:

chosenObject = couchItemeleven;

objectChosen = true;

break;

case 27:

chosenObject = couchItemtwelve;

objectChosen = true;

break;

case 28:

chosenObject = couchItemthirteen;

objectChosen = true;

break;

case 29:

chosenObject = couchItemfourten;

objectChosen = true;

break;

case 30:

chosenObject = couchItemfifteen;

objectChosen = true;

break;

//Cases Table

case 31:

chosenObject = tableItemOne;

objectChosen = true;

break;

case 32:

chosenObject = tableItemTwo;

objectChosen = true;

break;

case 33:

chosenObject = tableItemThree;

objectChosen = true;

break;

case 34:

chosenObject = tableItemFour;

objectChosen = true;

break;

case 35:

chosenObject = tableItemFive;

objectChosen = true;

break;

case 36:

chosenObject = tableItemSix;

objectChosen = true;

break;

case 37:

chosenObject = tableItemseven;

objectChosen = true;

break;

case 38:

chosenObject = tableItemeight;

objectChosen = true;

break;

case 39:

chosenObject = tableItemnine;

objectChosen = true;

break;

case 40:

chosenObject = tableItemten;

objectChosen = true;

break;

case 41:

chosenObject = tableItemeleven;

objectChosen = true;

break;

case 42:

chosenObject = tableItemtwelve;

objectChosen = true;

break;

case 43:

chosenObject = tableItemthirteen;

objectChosen = true;

break;

case 44:

chosenObject = tableItemfourten;

objectChosen = true;

break;

case 45:

chosenObject = tableItemfifteen;

objectChosen = true;

break;

//Cases 19-24 Bed.

case 46:

chosenObject = bedOne;

objectChosen = true;

break;

case 47:

chosenObject = bedTwo;

objectChosen = true;

break;

case 48:

chosenObject = bedThree;

objectChosen = true;

break;

case 49:

chosenObject = bedFour;

objectChosen = true;

break;

case 50:

chosenObject = bedFive;

objectChosen = true;

break;

case 51:

chosenObject = bedSix;

objectChosen = true;

break;

case 52:

chosenObject = bedseven;

objectChosen = true;

break;

case 53:

chosenObject = bedeight;

objectChosen = true;

break;

case 54:

chosenObject = bednine;

objectChosen = true;

break;

case 55:

chosenObject = bedten;

objectChosen = true;

break;

case 56:

chosenObject = bedeleven;

objectChosen = true;

break;

case 57:

chosenObject = bedtwelve;

objectChosen = true;

break;

case 58:

chosenObject = bedthirteen;

objectChosen = true;

break;

case 59:

chosenObject = bedfourten;

objectChosen = true;

break;

case 60:

chosenObject = bedfifteen;

objectChosen = true;

break;

//Cases 61-75 devices.

case 61:

chosenObject = devicesOne;

objectChosen = true;

break;

case 62:

chosenObject = devicesTwo;

objectChosen = true;

break;

case 63:

chosenObject = devicesThree;

objectChosen = true;

break;

case 64:

chosenObject = devicesFour;

objectChosen = true;

break;

case 65:

chosenObject = devicesFive;

objectChosen = true;

break;

case 66:

chosenObject = devicesSix;

objectChosen = true;

break;

case 67:

chosenObject = devicesseven;

objectChosen = true;

break;

case 68:

chosenObject = deviceseight;

objectChosen = true;

break;

case 69:

chosenObject = devicesnine;

objectChosen = true;

break;

case 70:

chosenObject = devicesten;

objectChosen = true;

break;

case 71:

chosenObject = deviceseleven;

objectChosen = true;

break;

case 72:

chosenObject = devicestwelve;

objectChosen = true;

break;

case 73:

chosenObject = devicesthirteen;

objectChosen = true;

break;

case 74:

chosenObject = devicesfourten;

objectChosen = true;

break;

case 75:

chosenObject = devicesfifteen;

objectChosen = true;

break;

//Cases 76-90 healthytools.

case 76:

chosenObject = healthytoolsOne;

objectChosen = true;

break;

case 77:

chosenObject = healthytoolsTwo;

objectChosen = true;

break;

case 78:

chosenObject = healthytoolsThree;

objectChosen = true;

break;

case 79:

chosenObject = healthytoolsFour;

objectChosen = true;

break;

case 80:

chosenObject = healthytoolsFive;

objectChosen = true;

break;

case 81:

chosenObject = healthytoolsSix;

objectChosen = true;

break;

case 82:

chosenObject = healthytoolsseven;

objectChosen = true;

break;

case 83:

chosenObject = healthytoolseight;

objectChosen = true;

break;

case 84:

chosenObject = healthytoolsnine;

objectChosen = true;

break;

case 85:

chosenObject = healthytoolsten;

objectChosen = true;

break;

case 86:

chosenObject = healthytoolseleven;

objectChosen = true;

break;

case 87:

chosenObject = healthytoolstwelve;

objectChosen = true;

break;

case 88:

chosenObject = healthytoolsthirteen;

objectChosen = true;

break;

case 89:

chosenObject = healthytoolsfourten;

objectChosen = true;

break;

case 90:

chosenObject = healthytoolsfifteen;

objectChosen = true;

break;

default:

Debug.Log("Ei voi tapahtua");

break;

}

}

private void DeleteObject()

{

if (spawnedObject != null)

{

Destroy(spawnedObject);

chosenObject = null;

deleteButton.gameObject.SetActive(false);

colorButton.gameObject.SetActive(false);

//LeftRotate.gameObject.SetActive(false);

//RightRotate.gameObject.SetActive(false);

}

}

}