### **Задачи-эксперименты**

### **Задача 1. Отображение текстового объекта на маркере**

Создайте страницу, где вместо куба на маркере будет отображаться текст. Настройте текст (цвет, размер, положение), чтобы он появлялся над маркером.

**Подсказка**: Используйте тег <a-text>:

<a-text value="Привет, AR!" position="0 0.5 0" color="blue"></a-text>

### **Задача 2. Добавление нескольких объектов**

Сделайте сцену, где на маркере будет отображаться несколько объектов одновременно (например, куб, сфера и конус). Разместите их на разной высоте.

**Требования**:

* Куб красного цвета.
* Сфера синего цвета.
* Конус зеленого цвета.

**Подсказка**: Используйте теги <a-box>, <a-sphere>, <a-cone>.

### **Задача 3. Использование собственного маркера**

Создайте страницу, где вместо предустановленного маркера "hiro" будет использоваться ваш собственный маркер (например, QR-код).

**Подсказка**:

1. Создайте маркер на сайте https://jeromeetienne.github.io/AR.js/marker-training/examples/generator.html.
2. Укажите путь к маркеру с помощью атрибута type и url:  
   <a-marker type="pattern" url="path/to/your/marker.patt">

### **Задача 4. Создание анимации объекта**

Сделайте так, чтобы куб на маркере вращался вокруг своей оси и менял цвет.

**Подсказка**: Добавьте компонент animation в тег <a-box>:

<a-box position="0 0.5 0" color="red"

animation="property: rotation; to: 0 360 0; loop: true; dur: 5000"

animation\_\_color="property: material.color; to: blue; loop: true; dur: 2000">

</a-box>

### **Задача 5. Взаимодействие с объектами**

Сделайте объект на маркере интерактивным: при клике он должен изменять форму (например, из куба в сферу) или цвет.

**Подсказка**:

Добавьте слушатель событий click:  
<a-box position="0 0.5 0" color="red" id="interactiveBox"></a-box>

<script>

document.querySelector('#interactiveBox').addEventListener('click', function () {

this.setAttribute('geometry', 'primitive: sphere');

this.setAttribute('material', 'color: green');

});

</script>

### **Задача 6. Добавление 3D-модели из внешнего источника**

Добавьте в сцену 3D-модель (например, формата .glb или .gltf) вместо стандартного куба. Используйте бесплатные 3D-модели из [Sketchfab](https://sketchfab.com) или [Poly Haven](https://polyhaven.com).

**Подсказка**:

<a-entity

gltf-model="url(path/to/model.glb)"

position="0 0 0" scale="0.5 0.5 0.5"></a-entity>

### **Задача 7. Создание нескольких маркеров**

Создайте сцену с несколькими маркерами, где каждый отображает разные объекты. Например, один маркер показывает куб, а другой — сферу.

**Подсказка**:

<a-marker preset="hiro">

<a-box position="0 0.5 0" color="red"></a-box>

</a-marker>

<a-marker preset="kanji">

<a-sphere position="0 0.5 0" color="blue"></a-sphere>

</a-marker>

### **Задача 8. Реализация анимации маркера**

Сделайте так, чтобы объект на маркере анимировался, только когда маркер находится в поле зрения камеры (например, вращался).

**Подсказка**: Добавьте анимацию:

<a-animation

attribute="rotation"

to="0 360 0"

dur="2000"

repeat="indefinite"></a-animation>

### **Задача 9. Изменение объекта по времени**

Создайте сцену, где объект меняет свою форму (например, из куба в сферу) каждые 5 секунд.

**Подсказка**: Используйте JavaScript:

<script>

setInterval(() => {

const box = document.querySelector('#changingBox');

box.setAttribute('geometry', 'primitive: sphere');

setTimeout(() => {

box.setAttribute('geometry', 'primitive: box');

}, 5000);

}, 10000);

</script>

<a-box id="changingBox" position="0 0.5 0" color="green"></a-box>

### **Задача 10. Создание интерактивной кнопки**

Создайте AR-кнопку, которая при клике отображает новое сообщение или меняет цвет объекта.

**Подсказка**: Используйте событие click:

<a-box id="button" color="blue" position="0 0.5 0"></a-box>

<script>

document.querySelector('#button').addEventListener('click', () => {

alert('Кнопка нажата!');

});

</script>

### **Задача 11. Привязка AR-объектов к реальной геолокации**

Создайте AR-сцену, в которой 3D-объекты размещаются в определенных точках реального мира на основе координат GPS.

**Подсказка**: Используйте компонент gps-entity-place из AR.js:

<a-entity gps-entity-place="latitude: 37.7749; longitude: -122.4194;">

<a-box position="0 0.5 0" color="red"></a-box>

</a-entity>

### **Задача 12. Загрузка текстур на объект**

Создайте куб с текстурой (например, деревянной или кирпичной) вместо однотонного цвета.

**Подсказка**:

<a-box

position="0 0.5 0"

material="src: url(path/to/texture.jpg);"></a-box>

### **Задача 13. Создание AR-игры**

Создайте простую AR-игру, в которой пользователю нужно нажимать на появляющиеся в разных местах сферы, чтобы набрать очки.

**Подсказка**: Используйте JavaScript для изменения позиции сфер и учета очков.

### **Задача 14. Работа со светом**

Добавьте в сцену источник света, чтобы освещение объектов стало реалистичным. Экспериментируйте с разными типами света (точечный, направленный, окружной).

**Подсказка**:

<a-light type="point" position="2 4 3" intensity="1.5"></a-light>

### **Задача 15. Использование SLAM (без маркеров)**

Создайте сцену, где объекты располагаются в пространстве без использования маркеров, используя функцию SLAM в AR.js.

**Подсказка**: Включите атрибут arjs с режимом SLAM:

<a-scene embedded arjs="sourceType: webcam; debugUIEnabled: false;">

<a-box position="0 0.5 0" color="yellow"></a-box>

</a-scene>

### **Задача 16. Размещение текстовой информации**

Создайте объект, который сопровождается текстовым описанием. Например, 3D-модель планеты и её название.

**Подсказка**:

<a-text value="Earth" position="0 1 0" color="white"></a-text>

<a-sphere color="blue" radius="0.5"></a-sphere>

### **Задача 17. Динамическая подгрузка объектов**

Создайте сцену, где объекты загружаются только при приближении камеры.

**Подсказка**: Используйте JavaScript и отслеживайте положение камеры с помощью события.

### **Задача 18. Анимация камеры**

Создайте сцену, где камера плавно перемещается к объекту, когда пользователь кликает на него.

**Подсказка**: Используйте компонент animation для камеры:

<a-camera animation="property: position; to: 0 1 3; dur: 2000"></a-camera>

### **Задача 19. Включение AR-сцены по кнопке**

Создайте страницу, где AR-сцена появляется только после нажатия кнопки "Начать AR".

**Подсказка**:

<button onclick="document.querySelector('a-scene').style.display = 'block'">Начать AR</button>

<a-scene style="display: none;" embedded arjs>

<a-box position="0 0.5 0" color="green"></a-box>

</a-scene>

### **Задача 20. Воссоздание солнечной системы**

Создайте AR-модель солнечной системы с вращением планет вокруг солнца.

**Подсказка**: Используйте комбинацию компонентов animation и моделей a-sphere.