

Обзор библиотеки Unity

Кривцов Илья, 24.Б44-мм

Зачем нужен Unity?

- Игры
- Обучающие программы
- Рендер 3D графики
- И ещё много различных применений

Кратко про Unity

- Самый популярный движок для игр
- Кроссплатформенный
- Основной язык для скриптов — C#
- Но можно не писать код

Как писать код

- Используется C# 9
- Можно писать в любом текстовом редакторе / IDE
- Есть расширения для Visual Studio, VS Code, Rider
- Поддерживается отладка

Как писать код

```
public class MyScript : MonoBehaviour
{
    // самые важные методы
    void Start() { }
    void Update() { }

    // всё ещё важные методы
    void Awake() { }
    void FixedUpdate() { }
}
```

Как писать код

- `MonoBehaviour` — базовый класс для взаимодействия с объектами
- `Awake()` — вызывается сразу после создания компонента, можно (и нужно) использовать вместо конструктора
- `Start()` — вызывается сразу перед первым `Update()` объекта
- `Update()` — вызывается один раз за кадр
- `FixedUpdate()` — вызывается один раз за «физический» кадр; в отличие от `Update()` вызывается фиксированное количество раз в секунду (чаще всего 50, можно настроить)

ОСНОВНЫЕ ТИПЫ

// все игровые объекты на сцене

private GameObject obj;

// от него наследуются все компоненты

private Component component;

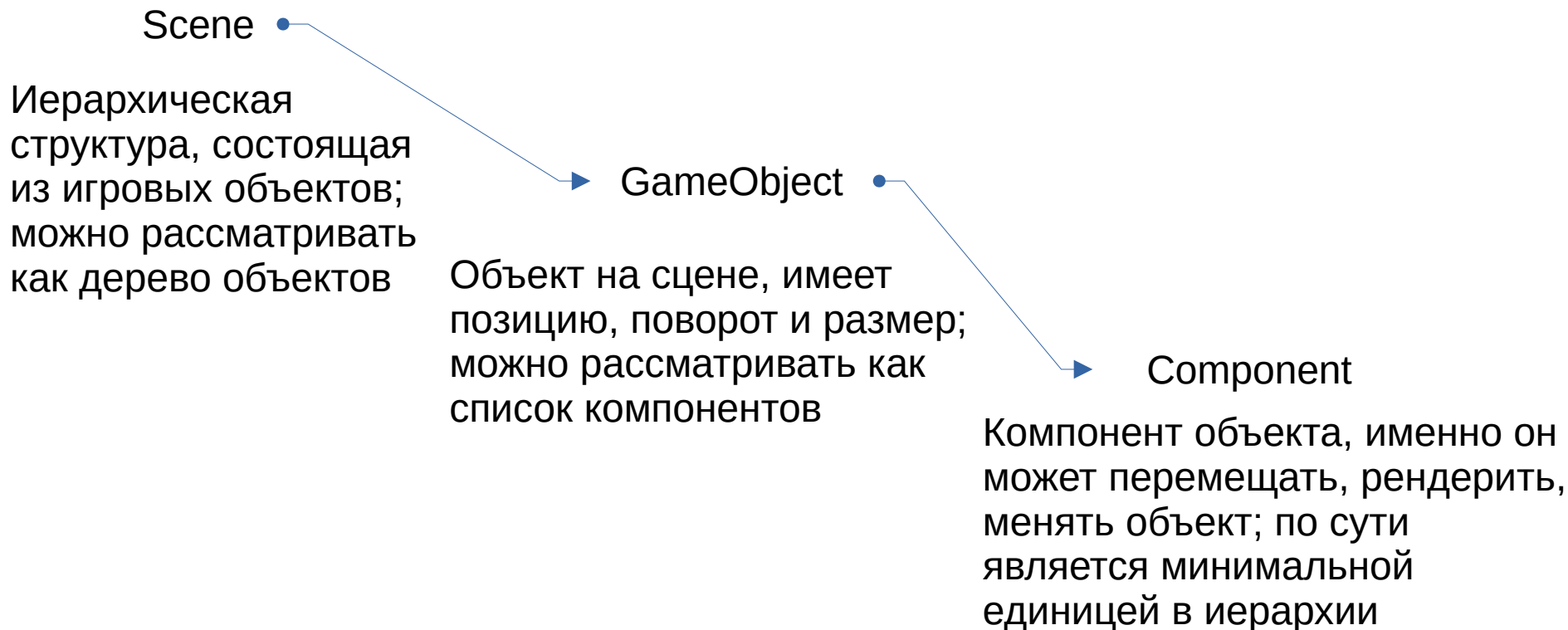
// для кастомных компонентов/скриптов

public class MyComponent : MonoBehaviour { }

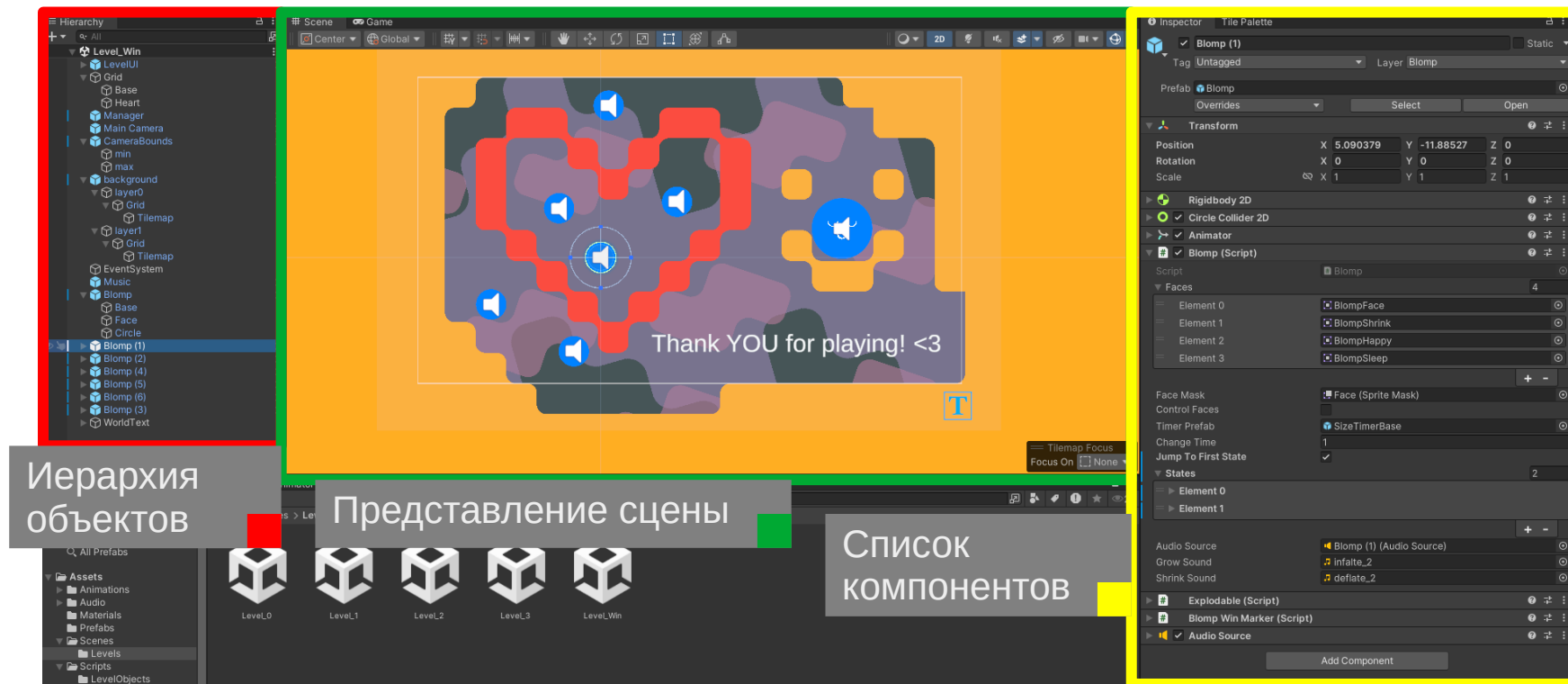
// обязательный для всех GameObject компонент

private Transform transform;

Иерархия на сцене



Иерархия на сцене



Корутины

```
private IEnumerator NpcInteraction()  
{  
    this.npc.Move(left);  
    yield return new WaitForSeconds(0.5f);  
    this.npc.Talk();  
    yield return new WaitForSeconds(1);  
    this.level.CompleteLevel();  
}
```

Корутины

- Корутина — метод, исполнение которого может прерываться и продолжаться
- `IEnumerator` — «бесплатная» машина состояний, создаваемая компилятором, можно использовать как корутину не только в Unity
- `WaitForSeconds` — ждёт указанное количество секунд
- `WaitForEndOfFrame` — ждёт следующего кадра
- `WaitWhile/WaitUntil` — ждут, пока указанный делегат не вернёт `false/true`