**Видимость объектов и ссылка null.**

[Java Syntax](https://javarush.ru/quests/QUEST_JAVA_SYNTAX)

[Уровень 6](https://javarush.ru/quests/lectures?quest=QUEST_JAVA_SYNTAX&level=6), Лекция 1

— Привет, Амиго!

— Привет, Элли! Что интересного расскажешь сегодня?

— Сегодня я расскажу тебе про время жизни объекта. Каждый объект после создания существует (живёт) пока хотя бы одна переменная хранит его адрес (на него есть хотя бы одна ссылка). Если ссылок больше не остаётся – объект умирает. Примеры:

public class MainClass

{

public static void main (String[] args)

{

┏ Tommy

┃ Cat cat = new Cat("Tommy");

┃ cat = null;

┗

┏ Sammy

┃ Cat cat1 = new Cat("Sammy");

┃┏ Maisy

┃┃Cat cat2 = new Cat("Maisy");

┃┃cat2 = cat1;

┃┗

┃┏ Ginger

┃┃cat1 = new Cat("Ginger");

┃┃ cat2 = null;

┃┗

┗

}

}

— Объект «кот Томми» существует всего одну строчку с момента создания. Уже на следующей строке единственную переменную, которая хранит на него ссылку, «обнуляют» и объект уничтожается Java-машиной.

— Объект «кот Семми» после создания хранится в переменной cat1. Или, если быть точным, там хранится ссылка на него. Парой строчек ниже эта ссылка копируется в cat2. После этого в cat1 сохраняется ссылка на другой объект, и ссылка на «кот Семми» остаётся только в cat2. Наконец, в последней строке метода main, последнюю ссылку на объект обнуляют.

— Объект «кот Мейси» существует сразу после создания всего одну строчку. На следующей строке переменной cat2 присваивают другое значение, и ссылка на «кот Мейси» теряется. Объект становится недостижимым и считается мусором (объект умер).

— Объект «кот Джинджер» существует сразу после создания и до конца метода. Вместе с окончанием метода будет уничтожена переменная cat1, и следом за ней и объект «кот Джинджер».

— Ясно.

— Но если создать объект Cat в каком-нибудь методе и сохранить ссылку на него в переменную класса, то объект Cat будет существовать все время, пока существует ссылка на него из другого живого объекта.

— Обычно объект уничтожается не сразу. Java-машина время от времени запускает уборку мусора – уничтожение неиспользуемых объектов. Об этом ниже.

— Если мы хотим, чтобы какая-то переменная перестала хранить ссылку на объект – можно присвоить ей значение null или же ссылку на другой объект.