1. Чем отличается класс от объекта? Приведите примеры и того, и другого, отличные от указанных в уроке.

Класс — это шаблон, описывающий свойства и методы, которыми будет обладать любой объект, созданный на основе этого класса.

Объект — это набор собранных внутри одной переменной свойств и/или функций. Каждое свойство представляет собой пару имя-значение.

Рецепт пирога это класс, а сам готовый пирог это объект.

1. Приведите примеры объектно-ориентированного подхода в реальной жизни.

Возможный вариант ответа: например, библиотека могла бы содержать классы Книга (название, автор, год выпуска, количество страниц), Сотрудники (ФИО, должность, зарплата, дата начала работы) и Читатели (ФИО, контакты, номер читательского билета).

Приведите примеры объектно-ориентированного подхода в программировании.

Возможный вариант ответа: например, товары в интернет магазине (название товара, тип, цена), комментарии в соцсети (автор комментария, дата, текст комментария), видео в Youtube (название видео, его адрес, автор и дата загрузки).

1. Что такое конструктор? Самостоятельно изучите и напишите, какие бывают виды конструкторов.

конструктор описывает шаблон, по которому будут создаваться объекты (экземпляры)

* конструктор с параметрами;
* [конструктор по умолчанию](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D0%BE%D1%80_%D0%BF%D0%BE_%D1%83%D0%BC%D0%BE%D0%BB%D1%87%D0%B0%D0%BD%D0%B8%D1%8E), не принимающий аргументов;
* именованный конструктор — функция, предполагающая явный вызов по имени, работающая как конструктор
* [конструктор копирования](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D0%BE%D1%80_%D0%BA%D0%BE%D0%BF%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) — конструктор, принимающий в качестве аргумента объект того же класса (или ссылку из него);
* конструктор преобразования — конструктор, принимающий один аргумент (эти конструкторы могут вызываться автоматически для преобразования значений других типов в объекты данного класса).
* конструктор перемещения

1. Что выведет код? Почему именно так?

function bike(){

    console.log(this.name)

}

let name = "ninja"

let obj1 ={name:"pomidor", bike: bike}

let obj2 ={name:"site", bike: bike}

bike()

obj1.bike()

obj2.bike()

код выдаст: pomidor и site

1. Чем статические свойства и методы отличаются от нестатических? В каких ситуациях они применяются?
2. Самостоятельно изучите, что такое геттеры и сеттеры, приведите пример класса с их использованием.
3. Что выведет код? Почему именно так?

class Person {

    constructor(name){

        this.name=name

    }

}

const member = new Person("John")

console.log(typeof member)

тип объекта возвращает объект

1. Что выведет код? Почему именно так?

const person = {

    name:"Valera",

    ags: 23

}

let city = person.city

city = "Amsterdam"

console.log(person)

Код выдаст свойства {name: 'Valera', ags: 23},