**Практическая работа**

**Задача 1: Создание класса.** Создайте класс Phone, который будет представлять телефон. У класса должны быть:

* Свойства: brand (бренд телефона), model (модель телефона) и batteryLevel (уровень заряда батареи в процентах, по умолчанию 100).
* Метод charge(), который увеличивает уровень заряда батареи до 100%.
* Метод useBattery(amount), который уменьшает уровень заряда батареи на указанное количество процентов.
* Метод info(), который выводит информацию о телефоне в формате: "Brand: <brand>, Model: <model>, Battery: <batteryLevel>%".

**Задача 2: Наследование.**

Создайте иерархию классов для устройств: Device (устройство), Phone (телефон) и Laptop (ноутбук). Каждое устройство имеет общие характеристики, такие как модель, производитель и год выпуска, но также обладает уникальными свойствами и методами.

1. **Базовый класс Device:**

Свойства:

* model (модель устройства)
* manufacturer (производитель)
* year (год выпуска)

Методы:

* getInfo() — возвращает строку с информацией об устройстве в формате: "Производитель: <manufacturer>, Модель: <model>, Год выпуска: <year>".

2. **Класс Phone**, наследуемый от Device:

super(model, manufacturer, year)

Дополнительные свойства:

* os (операционная система, например, "Android" или "iOS")
* isSmart (является ли телефон смартфоном, булевое значение)

- Методы:

* makeCall(number) — выводит в консоль строку: "Звонок на номер <number>...".
* getInfo() — переопределяет метод базового класса, добавляя информацию об операционной системе и типе телефона.

3. **Класс Laptop**, наследуемый от Device:

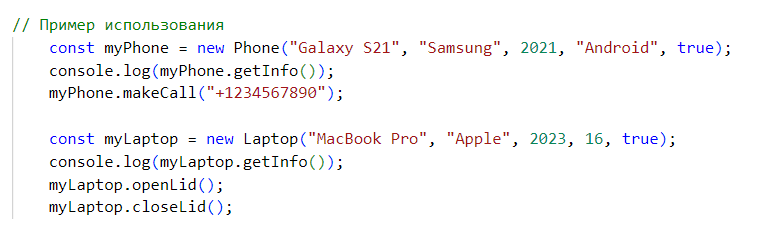
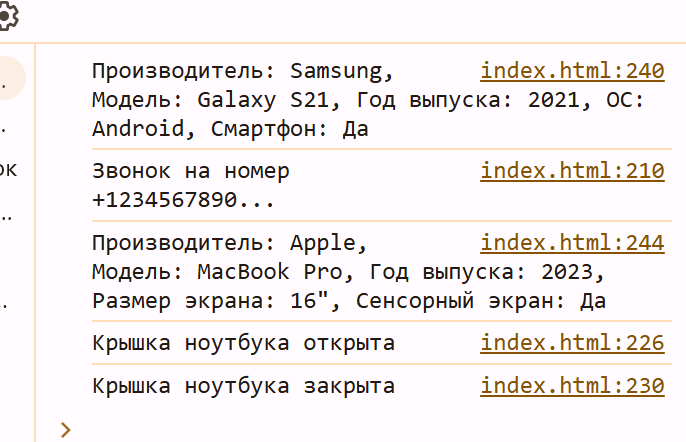
super(model, manufacturer, year)

Дополнительные свойства:

* screenSize (размер экрана в дюймах)
* isTouchScreen (является ли экран сенсорным, булевое значение)

Методы:

* openLid() — выводит в консоль строку: "Крышка ноутбука открыта".
* closeLid() — выводит в консоль строку: "Крышка ноутбука закрыта".
* getInfo() — переопределяет метод базового класса, добавляя информацию о размере экрана и типе экрана

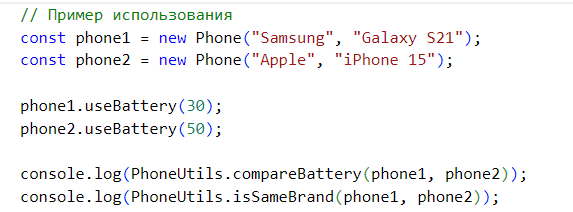
 

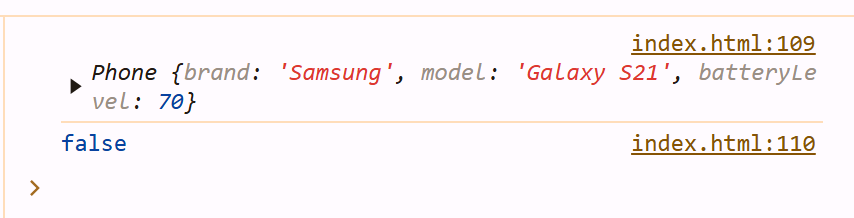
**Задача 3. Статические методы**

Создайте класс PhoneUtils, который будет содержать статические методы для работы с телефонами:

* compareBattery(phone1, phone2) — сравнивает уровень заряда батареи двух телефонов и возвращает телефон с более высоким уровнем заряда.
* isSameBrand(phone1, phone2) — проверяет, являются ли телефоны одного бренда.

Пример:





**Задача 4 Инкапсуляция (геттеры и сеттеры)**

Создайте класс Battery, который будет представлять батарею телефона. У класса должны быть:

* Приватное свойство #capacity (емкость батареи).
* Геттер capacity, который возвращает емкость батареи.
* Сеттер capacity, который устанавливает емкость батареи, только если новое значение больше 0. В противном случае выводит ошибку.
* Метод getBatteryLife(powerConsumption), который рассчитывает время работы батареи в часах на основе потребляемой мощности (в мАч/час). - (this.#capacity / powerConsumption)