

# Zoo App JS

## Домашнее задание: Создание системы управления зоопарком

### Инструкция:

Вам предстоит создать простую систему управления зоопарком с использованием объектно-ориентированного программирования в JavaScript. Система должна быть способна управлять разными типами животных в зоопарке. Вам нужно создать классы для следующих существ:

1. `Animal`: Базовый класс для всех животных. У него должны быть следующие атрибуты:
  - `name` (string): Имя животного.
  - `species` (string): Вид животного.
2. `Mammal`: Подкласс `Animal` для млекопитающих. У него должен быть дополнительный атрибут:
  - `furColor` (string): Цвет шерсти млекопитающего.
3. `Bird`: Подкласс `Animal` для птиц. У него должен быть дополнительный атрибут:
  - `wingSpan` (number): Размах крыльев птицы.
4. `Reptile`: Подкласс `Animal` для рептилий. У него должен быть дополнительный атрибут:
  - `scaleType` (string): Тип чешуи у рептилии.
5. `Zoo`: Класс для управления животными в зоопарке. У него должны быть следующие методы:
  - `addAnimal(animal)`: Добавить животное в зоопарк.
  - `listAnimals()`: Показать список всех животных в зоопарке.

- `getAnimalsBySpecies(species)` : Показать животных определенного вида в зоопарке.
- `removeAnimal(name)` : Удалить животное из зоопарка по имени.
- `countAnimals()` : Вернуть общее количество животных в зоопарке.

### Требования:

1. Создайте необходимые классы ( `Animal` , `Mammal` , `Bird` , `Reptile` И `Zoo` ) с их атрибутами и методами.
2. Реализуйте правильную инкапсуляцию, сделав атрибуты приватными и предоставив соответствующие геттеры и сеттеры.
3. Напишите программу `main.js` , которая демонстрирует функциональность системы управления зоопарком. Создайте несколько животных разных видов и выполняйте операции, такие как добавление, список и удаление животных из зоопарка.

### Бонусное задание (по желанию):

Создайте метод `feedAnimals()` в классе `Zoo` , который имитирует кормление всех животных в зоопарке. Каждый тип животного может иметь различный механизм кормления.