

# Classes and Inheritance

- We don't declare variables inside class with 'let' or 'const'
- we don't use 'function keyword' inside class functions
- prototype are default way of leveraging inheritance in JS
- we can see fn signature inside class instance prototype
- this = instance of class
- inheritance - extends keyword, child class will have all properties of parent class
- when creating child classes, super( ) method is required inside constructor of that child class
- Access Modifiers
  1. **public** = can be accessed outside class
  2. **protected** = within class or child class
  3. **private** = within parent class only
    - though child class can set private properties during instantiation
- **default** - public
- start everything with 'private', make them 'protected' or 'public' depending on the requirement
- explicitly declare access modifier for better code readability
- we can use shorter syntax for declaring in constructor functions
- we can specify 'get' and 'set' keyword for respective accessor functions for private properties
  - 'set' method can take only one parameter
  - use '\_' before private property
  - we can set, get private property just like any other property using ( . ) notation