

# Семинар №3

# Контекст

```
for (var i=3; i>0; i--) {  
    setTimeout(function() {  
        console.log(i);  
    }, 0)  
}
```

- **call**: func.call(context, arg1, arg2, ...);
- **apply**: func.apply(context, [arg1, arg2, ...]);
- **bind**: func.bind(context, [arg1], [arg2], ...);

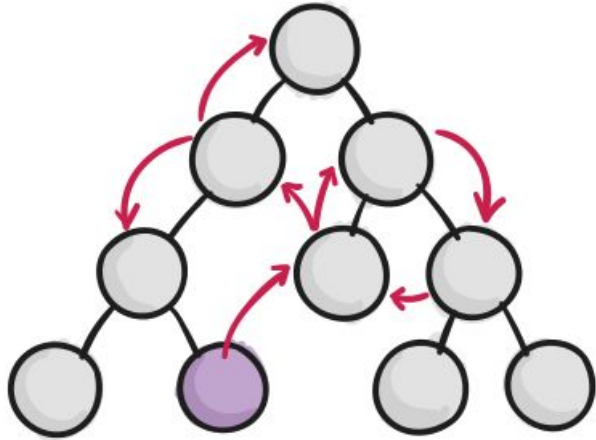
# Хотите разобраться в теме?

- Задачи, микрозадачи, очереди, планы
- `setTimeout`, `Promise` -- в каком порядке выполнятся?

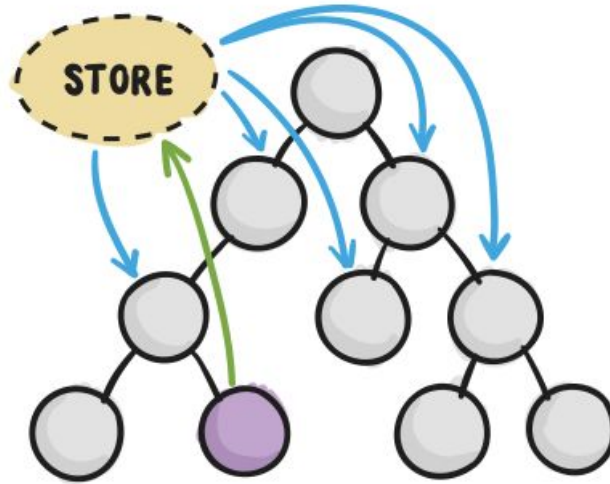
```
console.log('start');
setTimeout(function() {
  console.log(setTimeout);
}, 0);
Promise.resolve().then(function() {
  console.log('promise1');
}).then(function() {
  console.log('promise2');
});
console.log('end');
```

# npm install --save redux

WITHOUT REDUX



WITH REDUX



 COMPONENT INITIATING CHANGE