Hack your future BE

JAVASCRIPT 3

Everybody should learn to program a computer
Because it teaches you how to think

–STEVE JOBS

PROMISES

JAVASCRIPT IS SINGLE THREADED

THE ONLY BLOCKING FUNCTION HUMANS HAVE TO DEAL WITH IS SNEEZING

PROMISE ≃ EVENT LISTENER

FAIL/SUCCEED **ONCE**

ADD CALLBACK AFTER EVENT

=> CORRECT CALLBACK

PROMISE

STATES

- **Fulfilled** - The action relating to the promise succeeded
- **Rejected** - The action relating to the promise failed
- **Pending** - Hasn't fulfilled or rejected yet
- **Settled** - Has fulfilled or rejected

```
.catch(function(error) {
  console.log("Failed!", error);
})
```

```
.then(undefined, function(error) {
  console.log("Failed!", error);
})
```

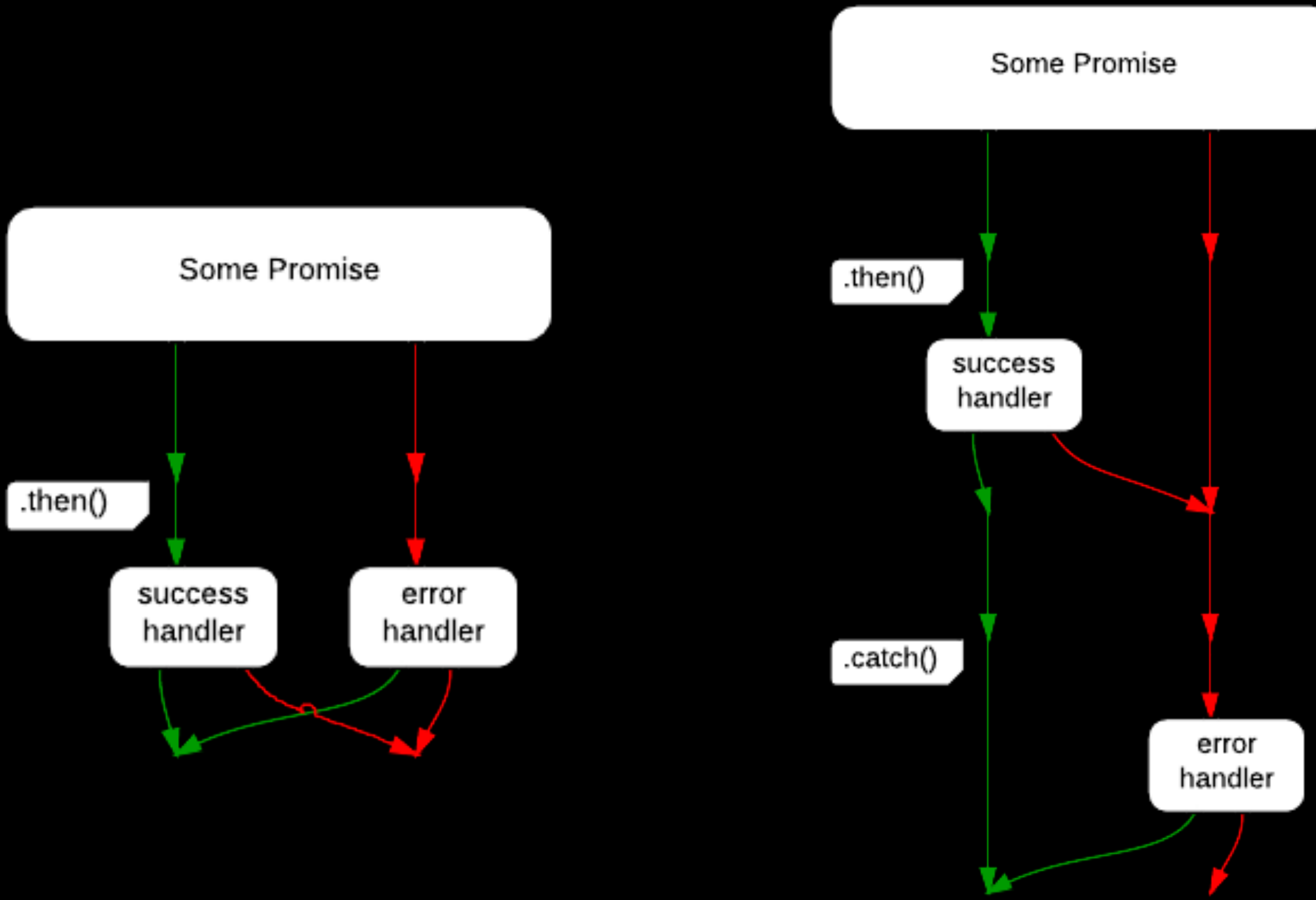
```
Const jsonPromise = new Promise(
 (resolve, reject) => {
   resolve(JSON.parse("JSON"));
 });
```

```
jsonPromise
 .then((data) => {
   console.log("It worked!", data);
 })
 .catch((err) => {
   console.log("It failed!", err);
 })
```

```
getJSON('story.json')
 .then((story) => {
   return getJSON(story.chapterUrls[0]);
 })
 .then((chapter1) => {
   addHtmlToPage(chapter1.html);
 })
 .catch(() => {
   addTextToPage("Failed to show chapter");
})
.then(() => {
   document.querySelector('.spinner')
   .style.display = 'none';
})
```

```
save()
 .then(
   handleSuccess,
   handleError
 );
```

```
save()
   .then(handleSuccess)
   .catch(handleError);
```


Some Promise
.then()
success handler
error handler
Some Promise
.then()
success handler
.catch()
error handler

END ALL PROMISE CHAINS WITH A

.CATCH()

PROMISE

.ALL()

returns a promise that resolves when **all of the promises**
in the argument have resolved,

or rejects with the reason of the first passed promise that rejects

PROMISE

.FINALLY()

runs when the promise is settled

resolve or reject

DEBUGGING

PROMISE EXERCISE 1

- Write two functions that use Promises that you can chain
- First function takes an array of words and capitalizes them
- Second function sorts the words in alphabetical order
- If the array contains anything but strings, it should throw an error

PROMISE EXERCISE 2

Loop through an array of URLs and fetch them

IN ORDER

HOMEWORK