

COMP1531

Week 7

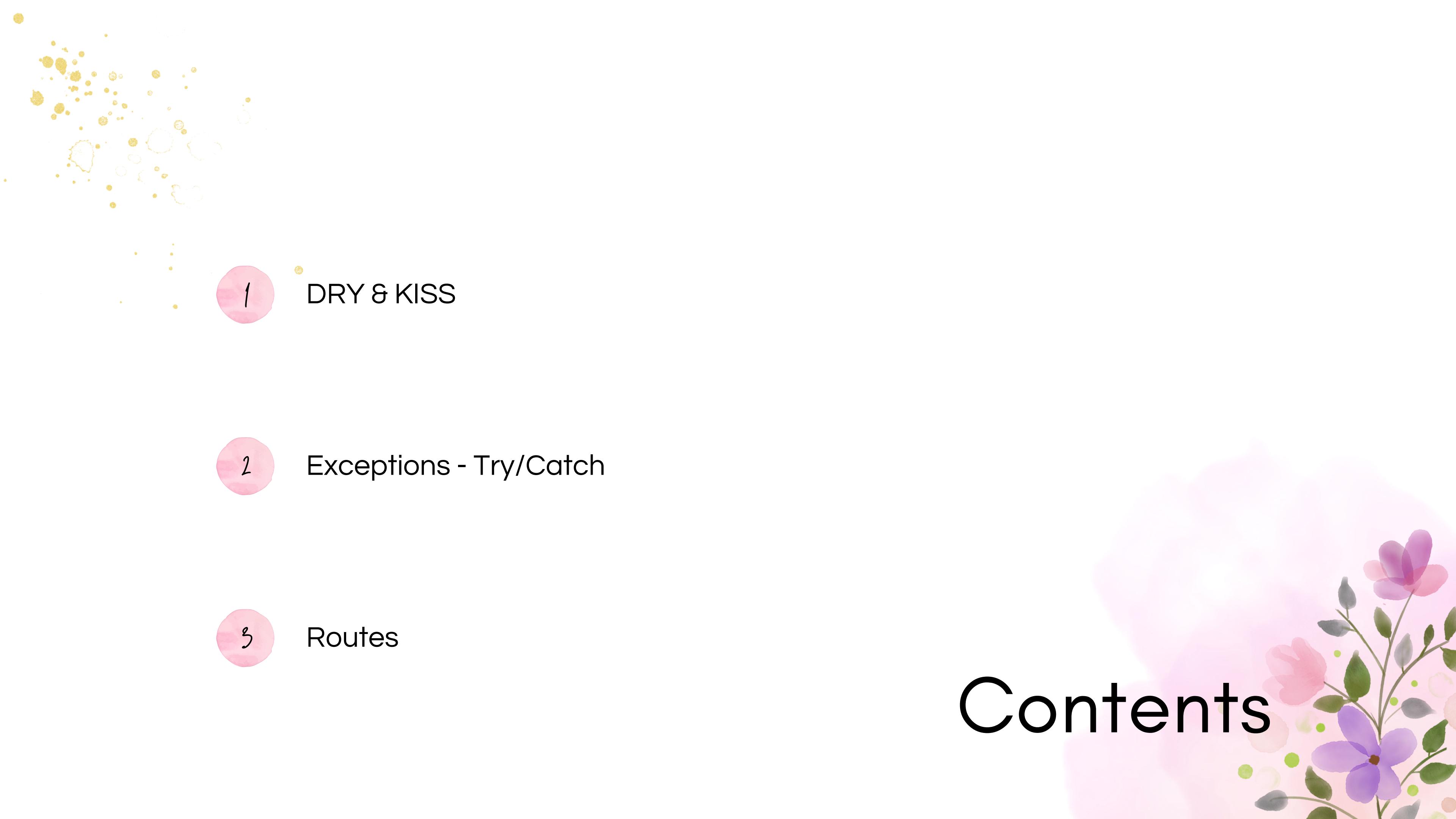
almost there!!!



Updates

- Iteration 2 Leaderboard
- Iteration 2 is due this week Friday 10pm
 - make sure all your code is in master
 - ensure you maintain good practices
 - code/tests
 - git
 - teamwork





1

DRY & KISS

2

Exceptions - Try/Catch

3

Routes

Contents





DRY

Don't
Repeat
Yourself

```
// Not DRY
function addNumbers(a, b) {
  console.log(a + b);
}

function subtractNumbers(a, b) {
  console.log(a - b);
}

// DRY
function calculateNumbers(a, b, operator) {
  if (operator === '+') {
    console.log(a + b);
  } else if (operator === '-') {
    console.log(a - b);
  }
}
```



KISS

Keep
It
Simple
Stupid



```
const numbers = [1, 2, 3, 4, 5];
const sum = numbers.reduce((total, num) => total + num,
  0);
```



Apply!

```
import promptSync from 'prompt-sync';

/**
 * Given an array of n integers, calculate the minimum, maximum, and the
 * product of the first n-1 numbers and last n-1 numbers.
 */
function drykiss(myList: number[]) {
    let myMin = Infinity;
    for (let i = 0; i < myList.length; i++) {
        if (myList[i] < myMin) {
            myMin = myList[i];
        }
    }

    let myMax = -Infinity;
    for (let i = 0; i < myList.length; i++) {
        if (myList[i] > myMax) {
            myMax = myList[i];
        }
    }

    let prod = 1;
    for (let i = 0; i < 4; i++) {
        prod = prod * myList[i];
    }
    const prodHead = prod;

    prod = 1;
    for (let i = 1; i < 5; i++) {
        prod = prod * myList[i];
    }

    const result = [myMin, myMax, prodHead, prod];
    return result;
}

const prompt = promptSync();
const a = parseInt(prompt('Enter a: '));
const b = parseInt(prompt('Enter b: '));
const c = parseInt(prompt('Enter c: '));
const d = parseInt(prompt('Enter d: '));
const e = parseInt(prompt('Enter e: '));
const myList = [a, b, c, d, e];
const result = drykiss(myList);
console.log('Minimum:');
console.log(` ${result[0]} `);
console.log('Maximum:');
console.log(` ${result[1]} `);
console.log('Product of first 4 numbers:');
console.log(` ${result[2]} `);
console.log('Product of last 4 numbers');
console.log(` ${result[3]} `);
```

Exceptions

Try / Catch

```
try {  
    nonExistentFunction();  
} catch (error) {  
    console.error(error);  
}
```

```
try {  
    throw "myException";  
} catch (err) {  
    logMyErrors(err);  
}
```

Routes



Take a look at a Youtube webpage. What routes do you think are necessary to allow this webpage to function? For each route:

- Discuss the data it might take in
- Discuss the data it might return



Routes

Are there any considerations that need to be made when choosing how to break up routes?

Finish Iteration 2!

there's no lab this week