# Empowering the Mind

@andrewjaykeller

AJ Keller

Co-founder at Neurosity

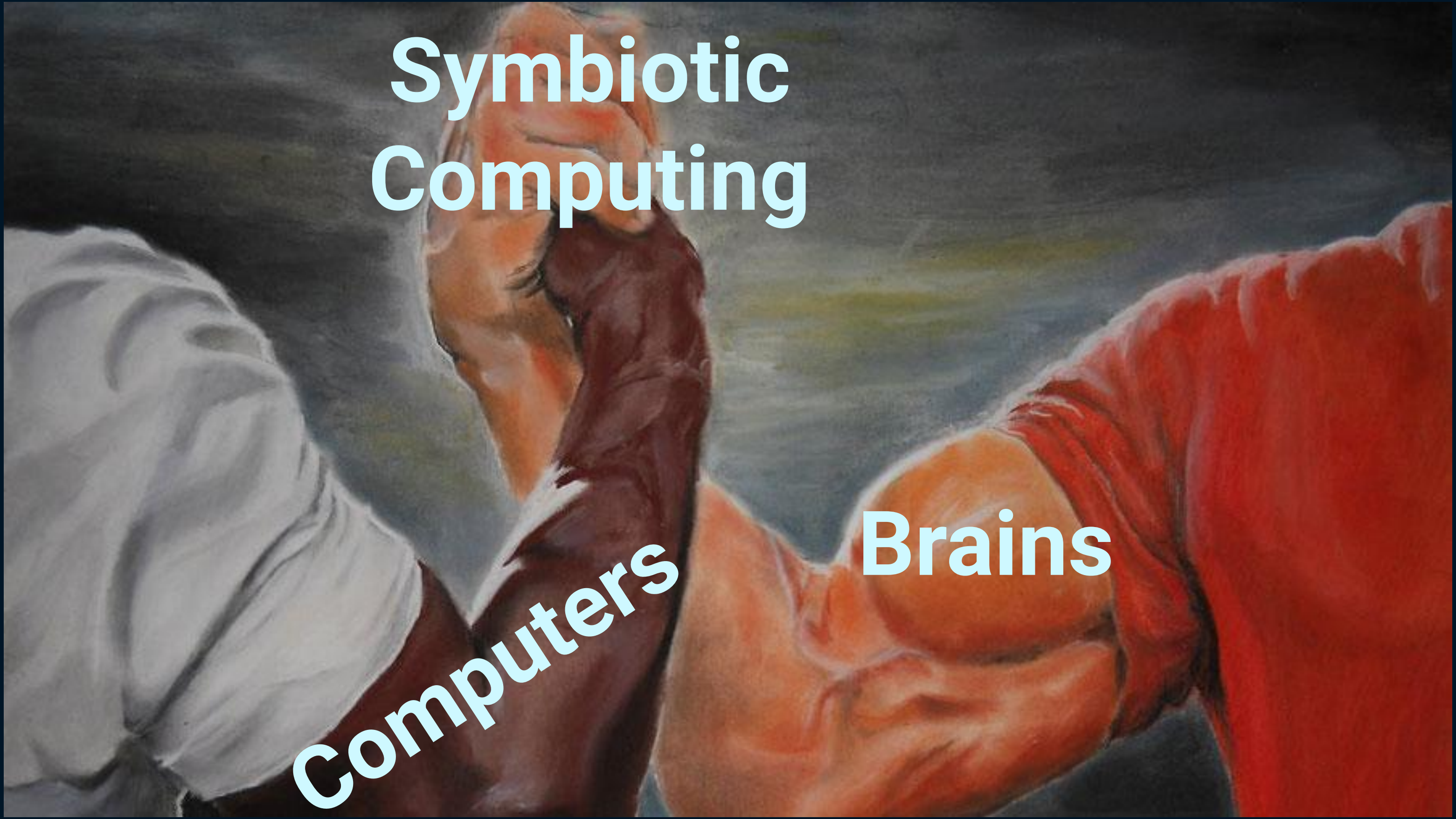
Lead Hardware Engineer

Follow me on twitter @andrewjaykeller

## Building User Adaptive Interfaces

- Symbiotic Computing
- Notion
- Notion API
- VSCode Extension
- Demo

## User Adaptive Interfaces


Symbiotic Computing
Computers
Brains

We believe in **empowering** the mind.

## Why now?

Micro Computers

Portable Brain Imaging

Machine Learning

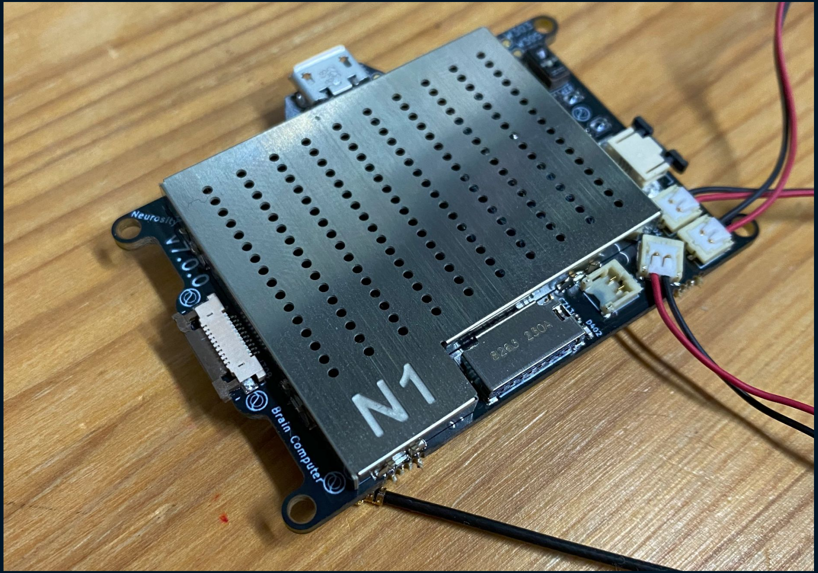
## Notion

## Neural **Device**

JavaScript API

On-board Machine Learning

@andrewjaykeller


N1
Neurosity
Brain Computer
B283 230A

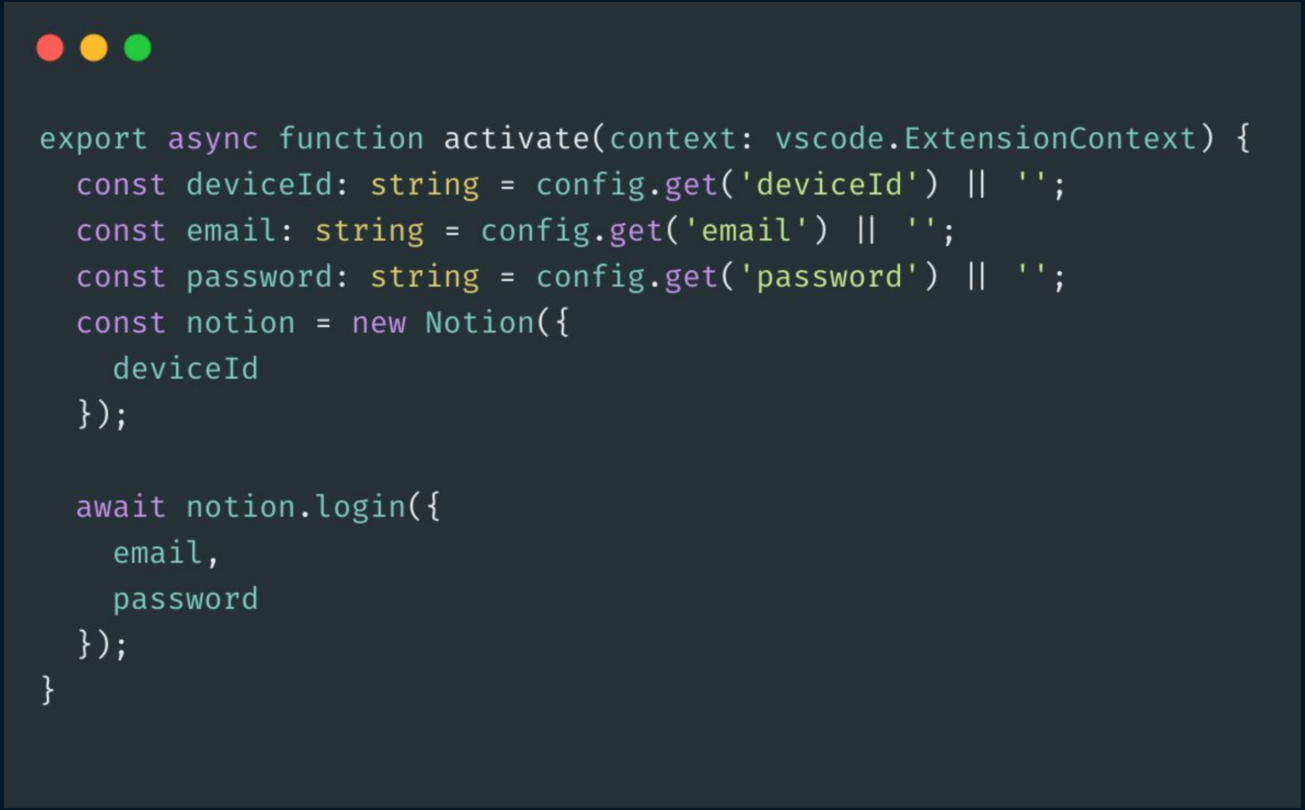
## Notion API

@andrewjaykeller

## Notion API

- Calm
- Focus
- Kinesis
- Channel Analysis
- Training
- Brainwaves
    - Raw
    - Frequency
    - PSD

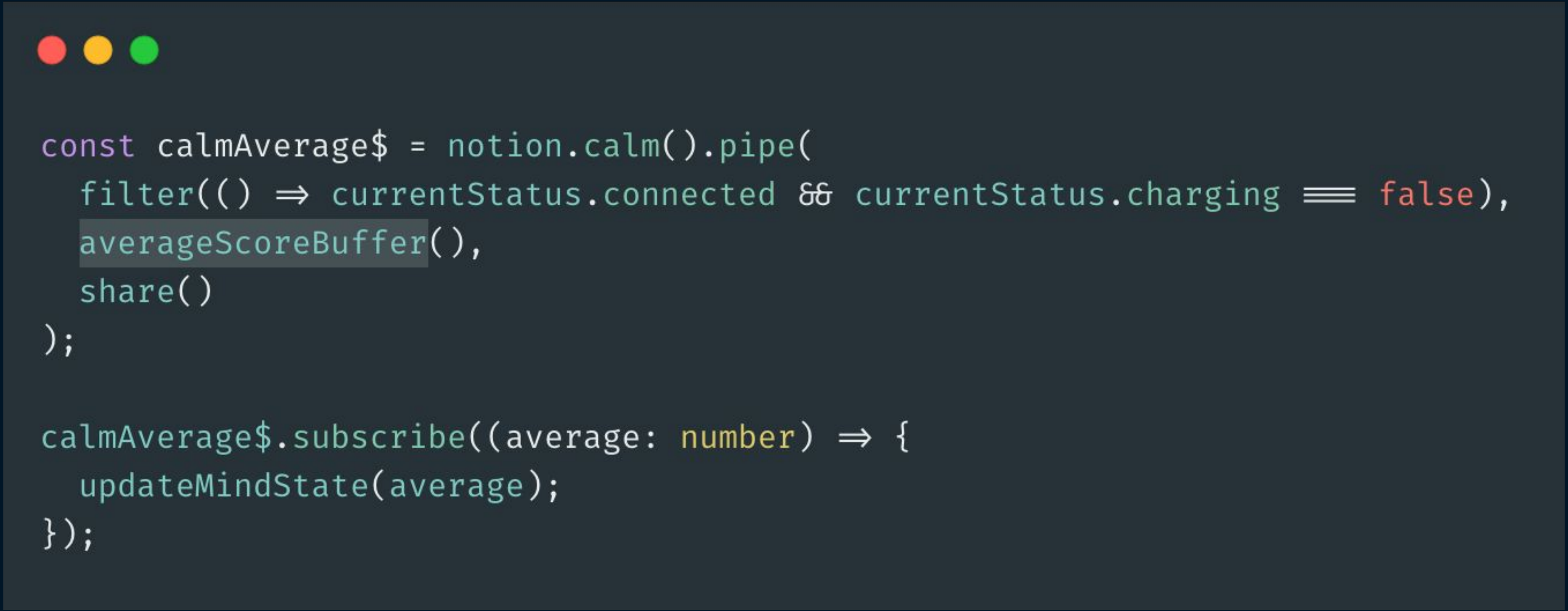
## VSCode Extension



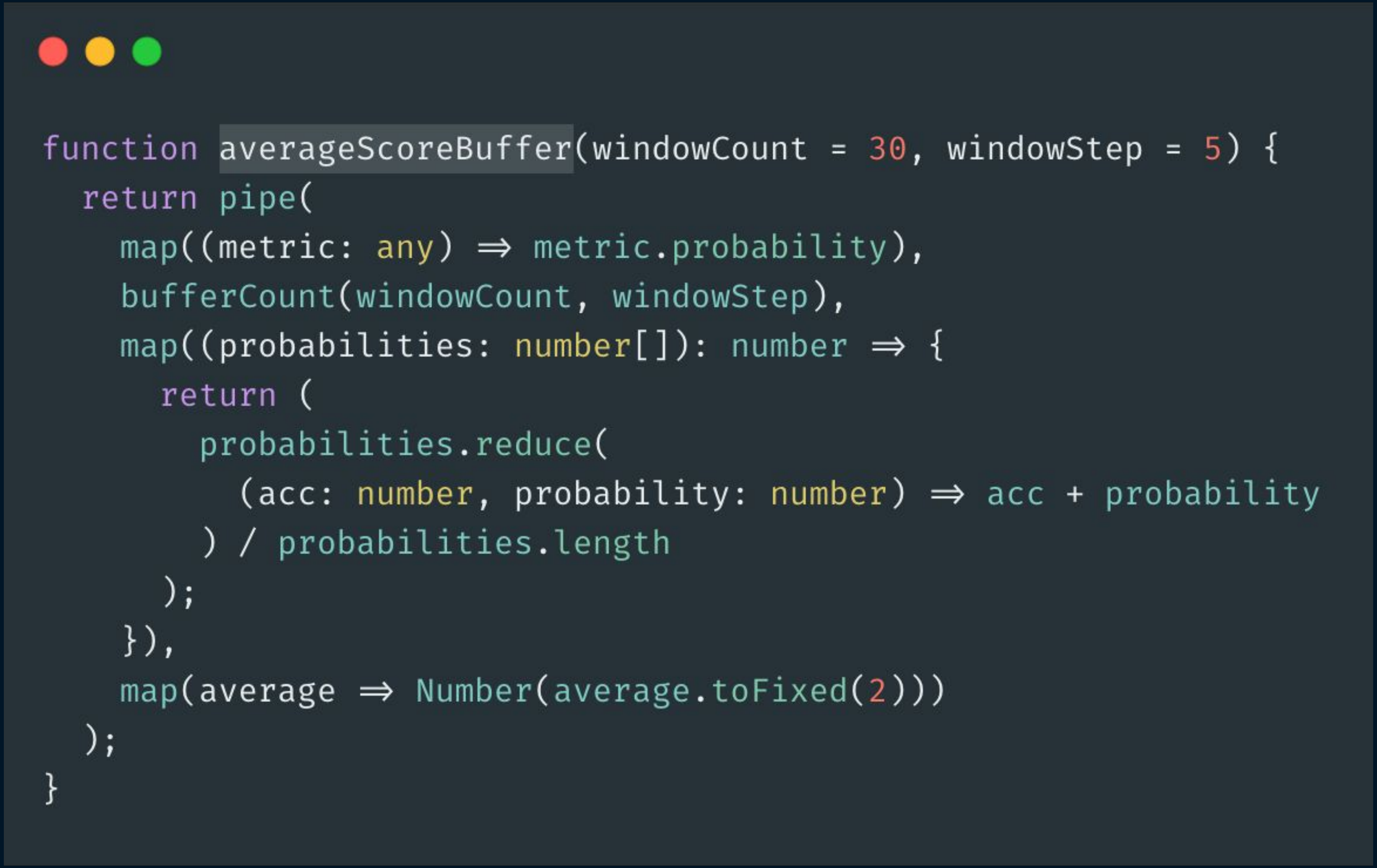
```
export async function activate(context: vscode.ExtensionContext) {
  const deviceId: string = config.get('deviceId') || '';
  const email: string = config.get('email') || '';
  const password: string = config.get('password') || '';
  const notion = new Notion({
    deviceId
  });

  await notion.login({
    email,
    password
  });
}
```

@andrewjaykeller

## Making Flow



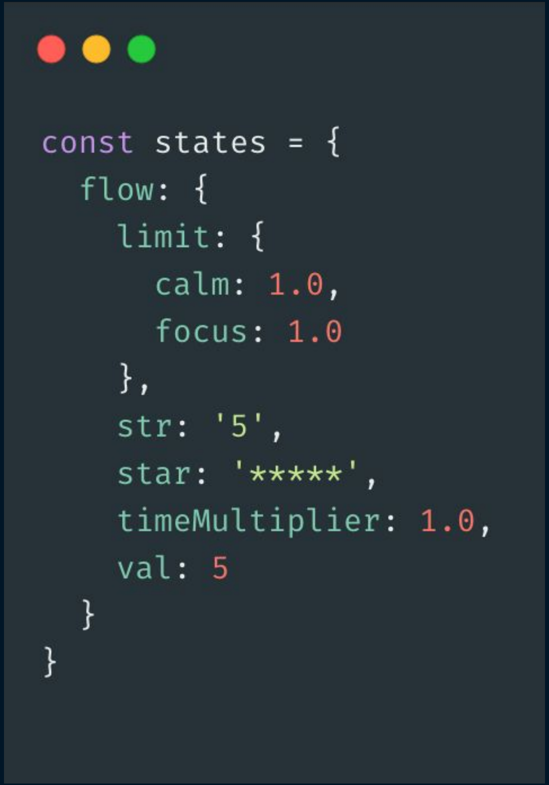
```
const calmAverage$ = notion.calm().pipe(
  filter(() => currentStatus.connected && currentStatus.charging === false),
  averageScoreBuffer(),
  share()
);

calmAverage$.subscribe((average: number) => {
  updateMindState(average);
});
```



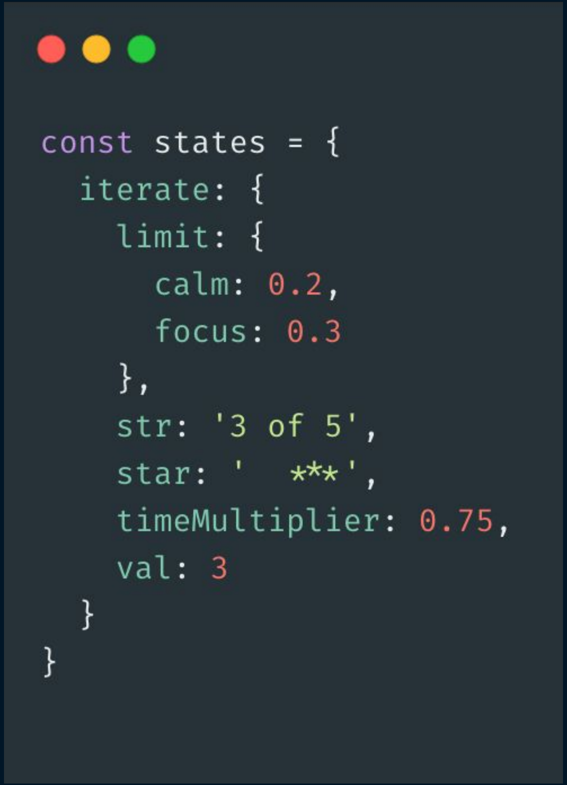
```typescript
function averageScoreBuffer(windowCount = 30, windowStep = 5) {
  return pipe(
    map((metric: any) => metric.probability),
    bufferCount(windowCount, windowStep),
    map((probabilities: number[]): number => {
      return (
        probabilities.reduce(
          (acc: number, probability: number) => acc + probability
        ) / probabilities.length
      );
    }),
    map(average => Number(average.toFixed(2)))
  );
}
```

@andrewjaykeller

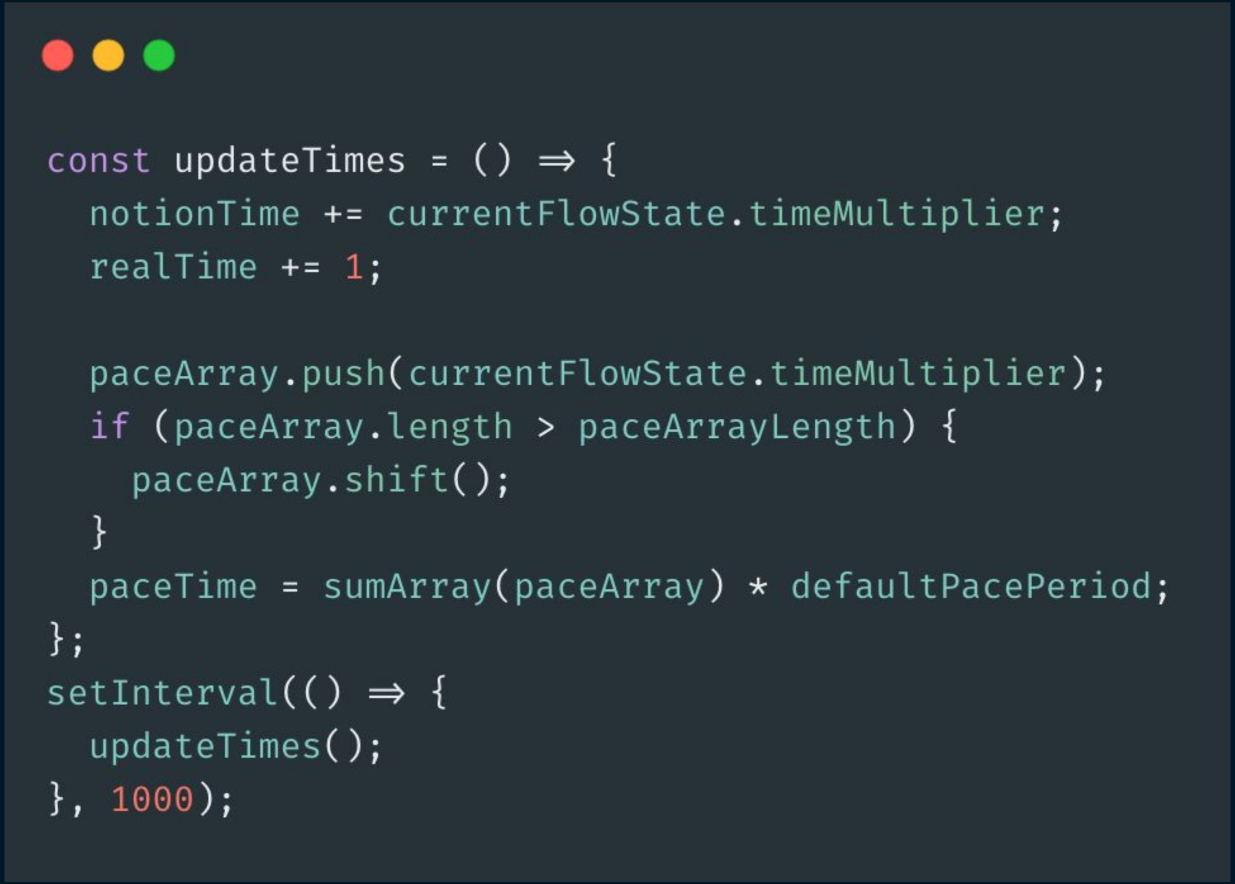
## Making time



```
const states = {
  flow: {
    limit: {
      calm: 1.0,
      focus: 1.0
    },
    str: '5',
    star: '*****',
    timeMultiplier: 1.0,
    val: 5
  }
}
```



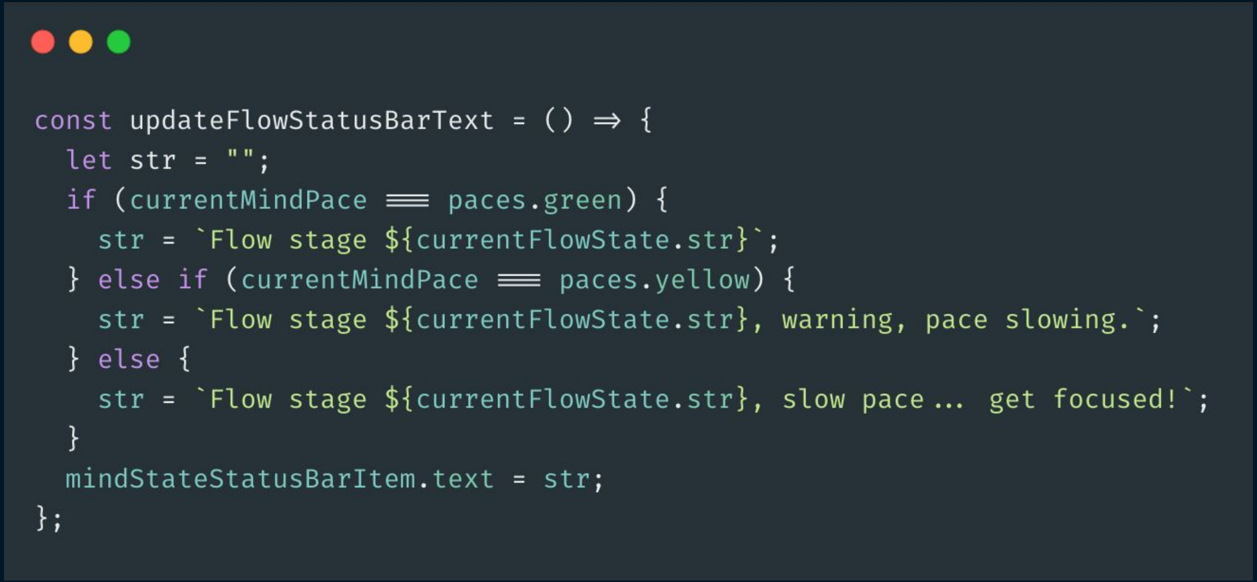
```
const states = {
  iterate: {
    limit: {
      calm: 0.2,
      focus: 0.3
    },
    str: '3 of 5',
    star: '  ***',
    timeMultiplier: 0.75,
    val: 3
  }
}
```



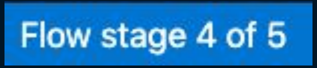
```javascript
const updateTimes = () => {
  notionTime += currentFlowState.timeMultiplier;
  realTime += 1;

  paceArray.push(currentFlowState.timeMultiplier);
  if (paceArray.length > paceArrayLength) {
    paceArray.shift();
  }
  paceTime = sumArray(paceArray) * defaultPacePeriod;
};
setInterval(() => {
  updateTimes();
}, 1000);
```

@andrewjaykeller

## Instant Feedback

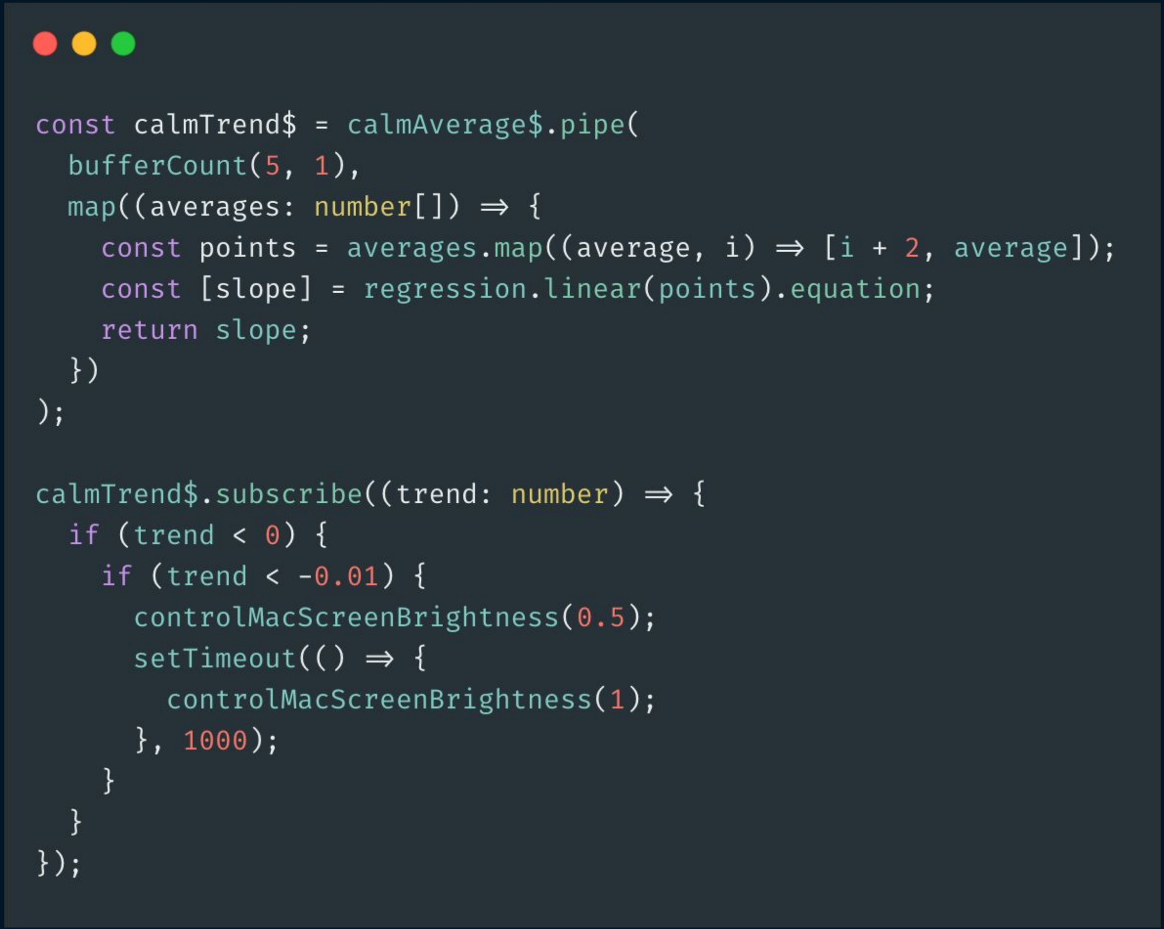


```javascript
const updateFlowStatusBarText = () => {
  let str = "";
  if (currentMindPace === paces.green) {
    str = `Flow stage ${currentFlowState.str}`;
  } else if (currentMindPace === paces.yellow) {
    str = `Flow stage ${currentFlowState.str}, warning, pace slowing.`;
  } else {
    str = `Flow stage ${currentFlowState.str}, slow pace... get focused!`;
  }
  mindStateStatusBarItem.text = str;
};
```



Flow stage 4 of 5

@andrewjaykeller



```
const calmTrend$ = calmAverage$.pipe(
  bufferCount(5, 1),
  map((averages: number[]) => {
    const points = averages.map((average, i) => [i + 2, average]);
    const [slope] = regression.linear(points).equation;
    return slope;
  })
);

calmTrend$.subscribe((trend: number) => {
  if (trend < 0) {
    if (trend < -0.01) {
      controlMacScreenBrightness(0.5);
      setTimeout(() => {
        controlMacScreenBrightness(1);
      }, 1000);
    }
  }
});
```