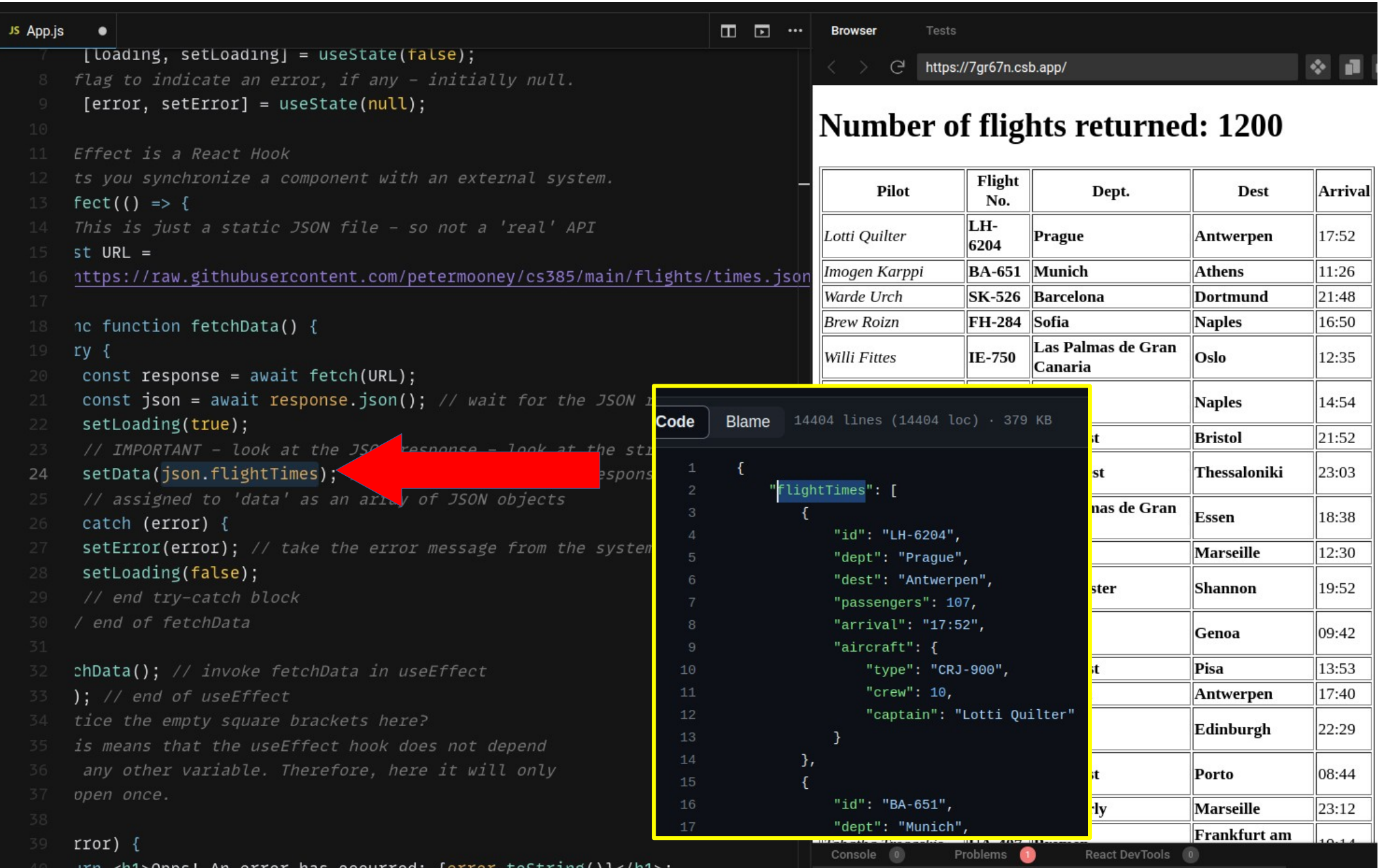
CS385 Lecture 10 – Working with API data in our applications

Situation – we now can obtain EXTERNAL data from an API

- **What can we do with the data within our React Application?**
- ANSWER – **Whatever we need to do with the data!**
- The JSON response, from the API, is now just a JSON object array.
- We can map, filter, reduce, sort, and so on. Essentially, we can perform any array-based operation we require.

Let's use `times.json` (from GitHub as our API data source)



Example 1: Filter by time

- Screenshot taken at 13:50 – we FILTER using the arrival property of the data

```
// the response of the API call from above.

function ResultsComponent(props) {
  // Creating a basic filter for times
  // for our purposes we can use very simple time comparison.
  // We assume all flights are TODAY

  function flightTimeFilter() {
    return function (flightObject) {
      // The arrival property of the data is HH:MM
      const timeParts = flightObject.arrival.split(":");
      let h = timeParts[0]; // extract the hour
      let m = timeParts[1]; // extract the minute
      // create a Javascript Date Object
      let flightDate = new Date();
      flightDate.setHours(h);
      flightDate.setMinutes(m);

      return flightDate >= new Date();
    };
  }

  // here we apply the filter directly to the API response
  let localAPIData = props.APIData.filter(flightTimeFilter());
  return (
    <>
      <h1>Number of flights returned: {localAPIData.length}</h1>
      <table border="1">
        <thead>
```

https://7gr67n.csb.app/

Number of flights returned: 740

Pilot	Flight No.	Dept.	Dest	Arrival
Lotti Quilter	**LH-6204**	**Prague**	**Antwerpen**	17:52
Warde Urch	**SK-526**	**Barcelona**	**Dortmund**	21:48
Brew Roizn	**FH-284**	**Sofia**	**Naples**	16:50
Celinda Grint	**PM-69853**	**Madrid**	**Naples**	14:54
Agace Patron	**CS-428**	**Budapest**	**Bristol**	21:52
Harlan Domino	**UA-5044**	**Bucharest**	**Thessaloniki**	23:03
Ruby Cochern	**FR-04515**	**Las Palmas de Gran Canaria**	**Essen**	18:38
Dar Swallwell	**UA-77749**	**Manchester**	**Shannon**	19:52
Devy Lally	**IE-9173**	**Bochum**	**Antwerpen**	17:40
Hobey Welbeck	**PM-4494**	**Tirana**	**Edinburgh**	22:29
Bernie Lindemann	**BA-865**	**Paris Orly**	**Marseille**	23:12
Tabatha Tregaskis	**UA-407**	**Bremen**	**Frankfurt am Main**	19:14
Carlie Cundict	**DK-2486**	**Amsterdam Schiphol**	**Krakow**	20:20
Christiana Valeri	**FH-6644**	**Belgrade**	**Zagreb**	18:35
Vale Larrie	**UA-673**	**Copenhagen**	**Antwerpen**	21:16
Leah Simmell	**IE-7725**	**Vienna**	**Zagreb**	18:10
Carlyn Foddy	**SY-39825**	**Sofia**	**Belfast**	16:17
Charmine Tringham	**LH-48361**	**Bucharest**	**Stuttgart**	15:06
Clotilda Alstead	**FH-24973**	**Vienna**	**Istanbul Atatürk**	20:20
Caresse Lowery	**SK-4030**	**Barcelona**	**Antwerpen**	23:22
Justis Struijs	**DK-6401**	**Milan**	**Oslo**	21:31
Meredith Olczyk	**FH-54237**	**Pristina**	**Genoa**	23:05

Example 2: Filter by dept/dest

App.js

```
  };
}

// A standard search for a string within the
// departure or destination property values
function flightDeptDestFilter(searchTerm) {
  return function (flightObject) {
    let dept = flightObject.dept.toLowerCase();
    let dest = flightObject.dest.toLowerCase();

    return (
      searchTerm != "" &&
      (dept.includes(searchTerm.toLowerCase()) ||
        dest.includes(searchTerm.toLowerCase()))
    );
  };
}

// here we apply the filter directly to the API response
let localAPIData = props.APIData.filter(flightDeptDestFilter("Sofia"));
return (
  <>
    <h1>Number of flights returned: {localAPIData.length}</h1>
    <table border="1">
      <thead>
        <tr>
          <th>Pilot</th>
          <th>Flight No.</th>
          <th>Dept.</th>
          <th>Dest</th>
          <th>Arrival</th>
        </tr>
      </thead>
```

Browser Tests

https://7gr67n.csb.app/

Number of flights returned: 24

Pilot	Flight No.	Dept.	Dest	Arrival
Brew Roizn	**FH-284**	**Sofia**	**Naples**	16:50
Carlyn Foddy	**SY-39825**	**Sofia**	**Belfast**	16:17
Malinda McCamish	**CS-444**	**Sofia**	**Thessaloniki**	08:38
Henka Rainbird	**LH-6737**	**Sofia**	**Turin**	23:28
Darda Roe	**SY-096**	**Sofia**	**Wroclaw**	13:18
Kaylyn Prewer	**SK-11726**	**Sofia**	**Krakow**	21:32
Cherry Poff	**SY-96306**	**Sofia**	**Riga**	20:30
Arliene Prudence	**UA-5528**	**Sofia**	**Zaragoza**	16:51
Kimmy Roskelley	**PM-01553**	**Sofia**	**Frankfurt am Main**	12:36
Emyle Winborn	**UA-3789**	**Sofia**	**Dublin**	09:23
Chris Ovitz	**FH-93961**	**Sofia**	**Goeteborg**	17:56
Hasty Ingall	**SK-617**	**Sofia**	**Naples**	15:11
Urban Tocqueville	**LH-05588**	**Sofia**	**Naples**	19:01
Janey Josephy	**BA-1781**	**Sofia**	**Riga**	13:48
Rudy Habbijam	**LH-4823**	**Sofia**	**Zagreb**	17:18
Paulie Behrens	**SK-98011**	**Sofia**	**Pisa**	09:44
Lenna Caps	**UA-8638**	**Sofia**	**Antwerpen**	22:40
Karlee Marnes	**IE-2861**	**Sofia**	**Dortmund**	11:54
Deni Antoszczyk	**UA-273**	**Sofia**	**Essen**	21:41
Cristal Glazebrook	**IE-22201**	**Sofia**	**Glasgow**	09:21
Rorie Tabbitt	**LH-469**	**Sofia**	**Frankfurt am Main**	09:30
Vinnie Lanchester	**IE-3660**	**Sofia**	**Sarajevo**	23:11
Quinlan MacGuiness	**FH-238**	**Sofia**	**Koeln**	14:04
Portia Trainer	**BA-1569**	**Sofia**	**Edinburgh**	10:13

Console 0 Problems 1 React DevTools 0

Ln 78, Col 46 Spaces: 2 UTF-8 LF JavaScri

Example 3- chain or compose MULTIPLE filters together

- Let's combine our time filter (screenshot 14:05) and our dept/dest filter

```
    );
  };
}

// here we apply the filter(s) directly to the API response
let localAPIData =
props.APIData.filter(flightDeptDestFilter("Sofia")).filter(flightTimeFilter());
return (
  <>
    <h1>Number of flights returned: {localAPIData.length}</h1>
    <table border="1">
      <thead>
        <tr>
          <th>Pilot</th>
          <th>Flight No.</th>
          <th>Dept.</th>
          <th>Dest</th>
          <th>Arrival</th>
        </tr>
      </thead>
      <tbody>
        {localAPIData.map((p, index) => (
          <tr key={index}>
            <td>
              <i>{p.aircraft.captain}</i>
```

https://7gr67n.csb.app/

Number of flights returned: 13

Pilot	Flight No.	Dept.	Dest	Arrival
Brew Roizn	**FH-284**	**Sofia**	**Naples**	16:50
Carlyn Foddy	**SY-39825**	**Sofia**	**Belfast**	16:17
Henka Rainbird	**LH-6737**	**Sofia**	**Turin**	23:28
Kaylyn Prewer	**SK-11726**	**Sofia**	**Krakow**	21:32
Cherry Poff	**SY-96306**	**Sofia**	**Riga**	20:30
Arliene Prudence	**UA-5528**	**Sofia**	**Zaragoza**	16:51
Chris Ovitz	**FH-93961**	**Sofia**	**Goeteborg**	17:56
Hasty Ingall	**SK-617**	**Sofia**	**Naples**	15:11
Urban Tocqueville	**LH-05588**	**Sofia**	**Naples**	19:01
Rudy Habbijam	**LH-4823**	**Sofia**	**Zagreb**	17:18
Lenna Caps	**UA-8638**	**Sofia**	**Antwerpen**	22:40
Deni Antoszczyk	**UA-273**	**Sofia**	**Essen**	21:41
Vinnie Lanchester	**IE-3660**	**Sofia**	**Sarajevo**	23:11

Challenges to think about for your project or larger apps

- Could you add a textbox to the application to allow someone to search the RESPONSE Data from the API?
- Could you write other types of filter functions (depending on the data)?
- How would you use multiple components – Rendering the output, obtaining user input, and so on?

Lecture 9 + 10 – THE END for Javascript coding today

Let’s talk about the project

- Example project screencasts from 2022/2023

CS385 Project Update

- **Are you working as a group (2, 3, or 4) or as an individual?**
- **YOU need to inform me [via email only]** of your project status before lectures on Tuesday 7th November 2023 (after mid-term break)
- One person per group can email.
- It is YOUR RESPONSIBILITY to inform me.

Let’s talk about Lab Exam 1