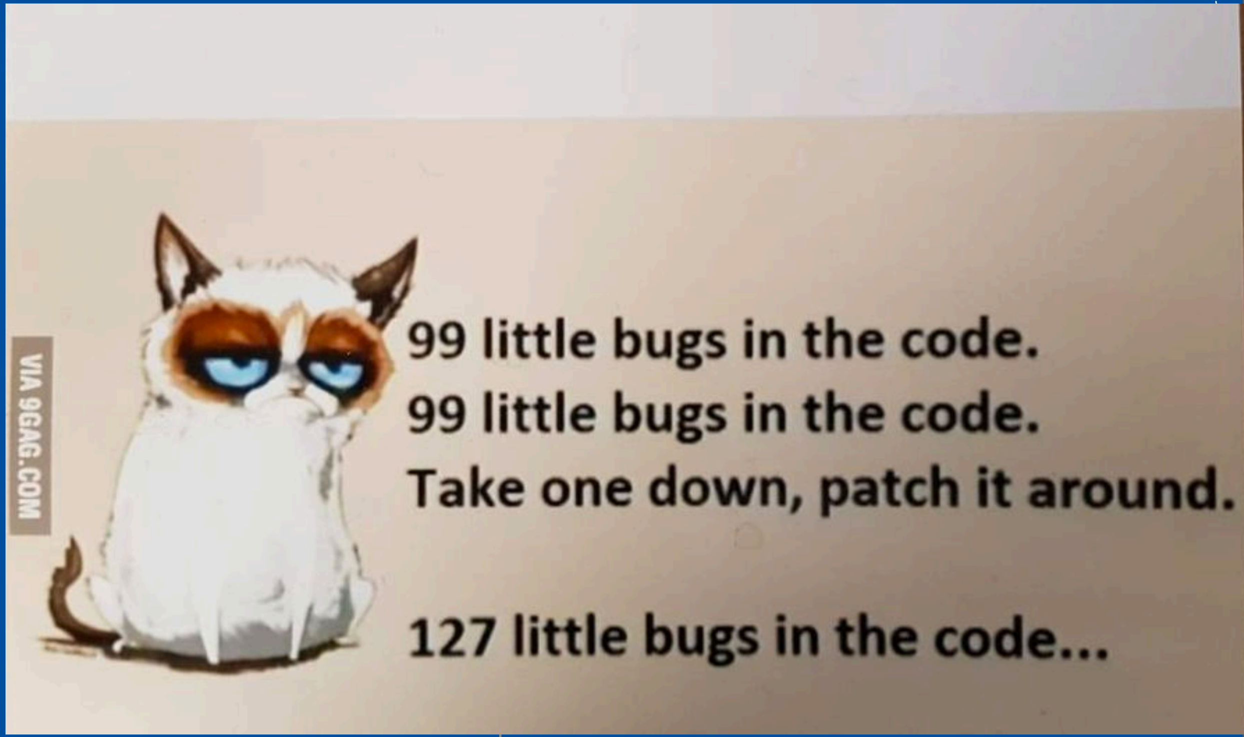
Week 1 Debugging NodeJS Applications

Prof. Fabio Ciravegna
Dipartimento di Informatica
Università di Torino
fabio.ciravegna@unito.it


VIA 9GAG.COM
99 little bugs in the code.
99 little bugs in the code.
Take one down, patch it around.
127 little bugs in the code...

Pease note

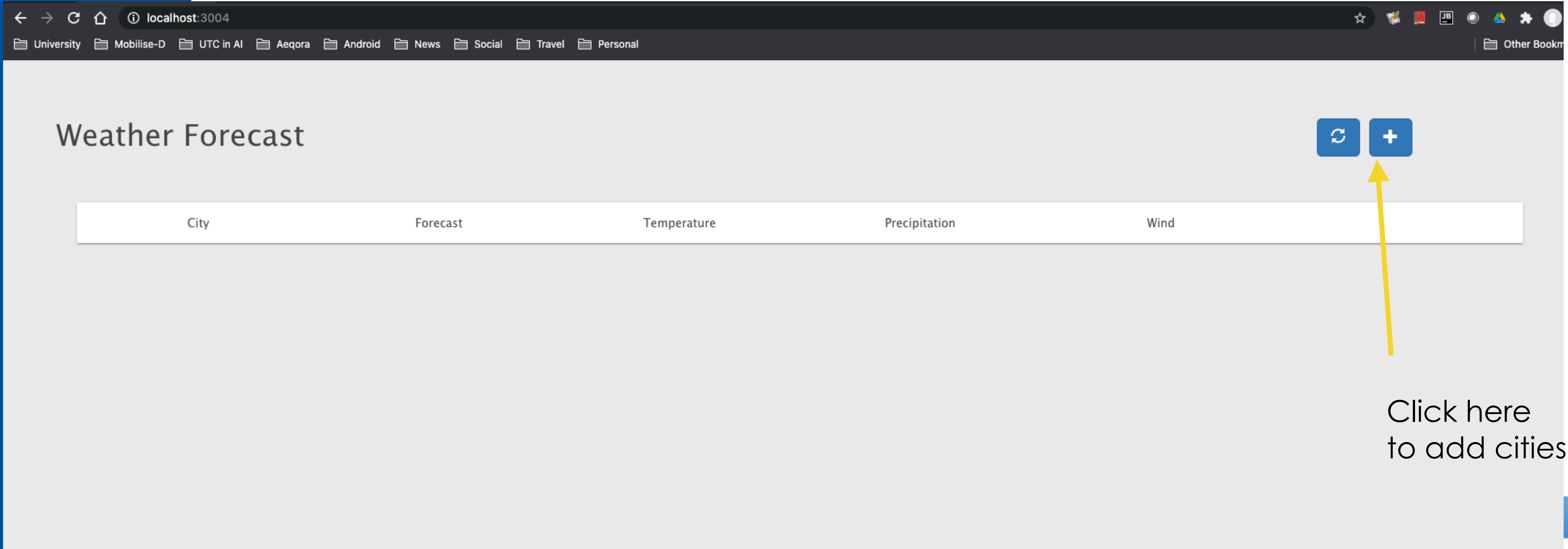
My screenshots of WebStorm may look different from yours mostly in the colours) because I use the high contrast settings of WebStorm (set under Preferences)

- You are given one rather large application
  - of which you do not need to understand much
    - it is the exercise for a following week
  - there are errors in it and you have to fix them
  - Errors are
    - Some spurious lines in the Javascript client
      - that you have to identify with Chrome's debugger
    - Some javascript errors in the server
      - that you have to identify with the IntelliJ debugger
      - action:
        - again just remove those lines

UNIVERSITÀ DI TORINO

this is what you will see

- Run the nodeJS project in IntelliJ
  - right click on bin/www
  - select run
- Open the browser on http://localhost:3004



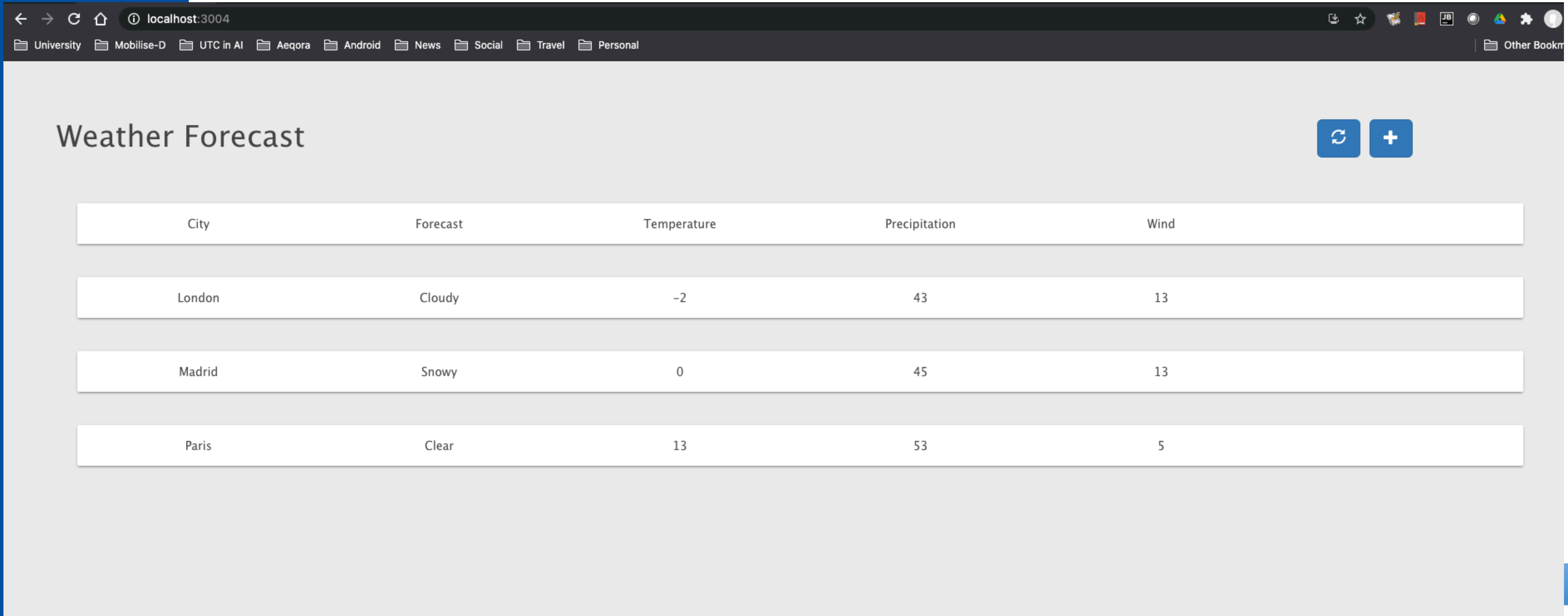
Weather Forecast

City	Forecast	Temperature	Precipitation	Wind

Click here to add cities

What you should see after adding some cities

- If there was no error

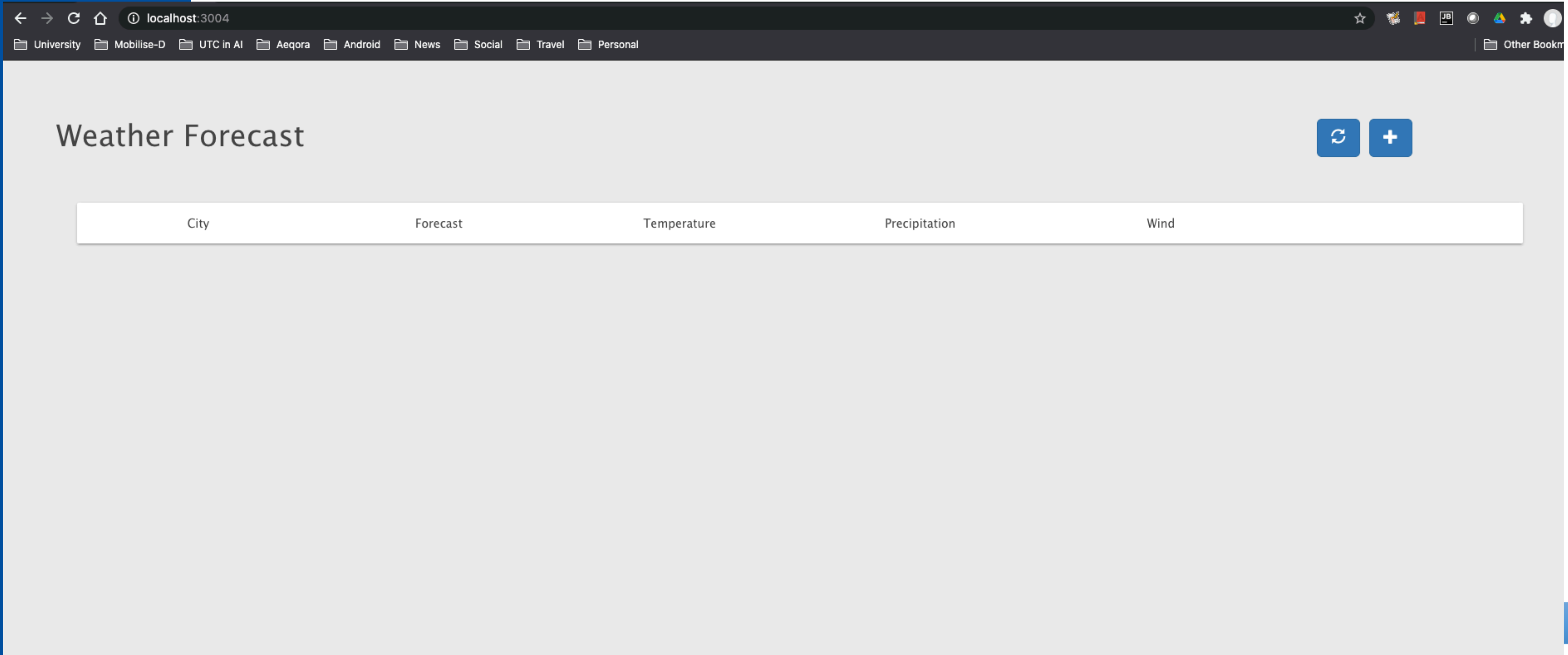


Weather Forecast

City	Forecast	Temperature	Precipitation	Wind
London	Cloudy	-2	43	13
Madrid	Snowy	0	45	13
Paris	Clear	13	53	5

Instead you will still see this

- No cities are added



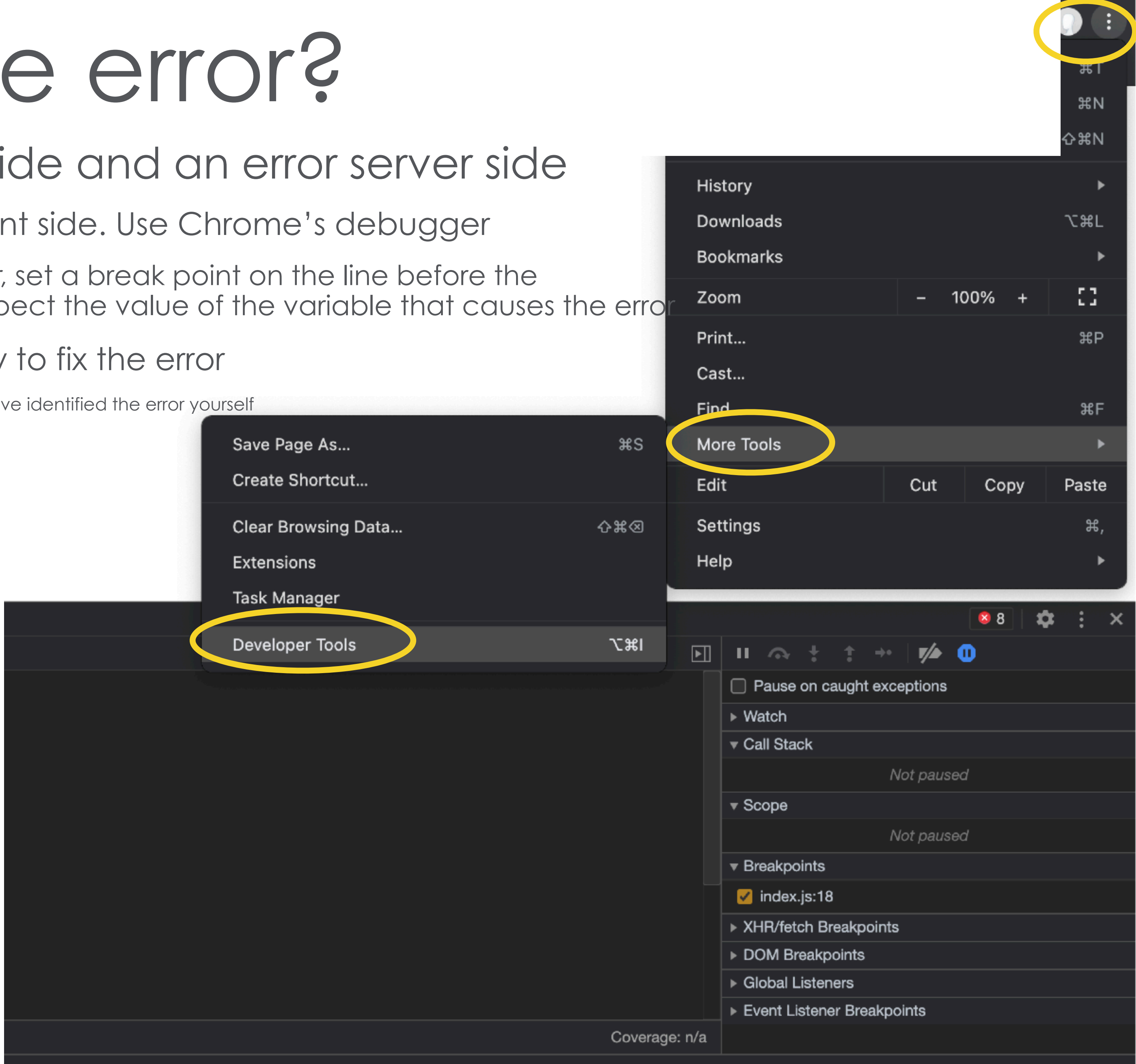
localhost:3004

Weather Forecast

City	Forecast	Temperature	Precipitation	Wind

Where is the error?

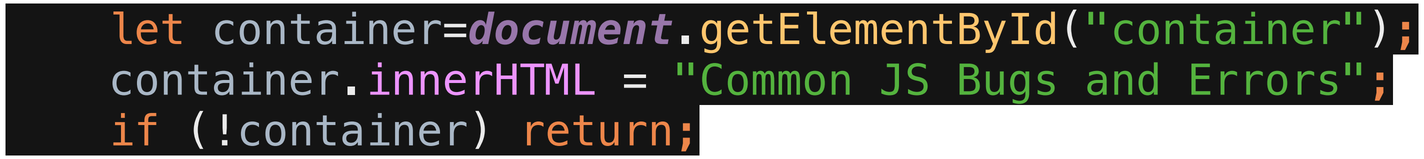
- There is an error client side and an error server side
  - Look first of all for the one client side. Use Chrome's debugger
    - Once you have found the error, set a break point on the line before the error. Reload the page and inspect the value of the variable that causes the error
  - See next slide to discover how to fix the error
    - **DO NOT LOOK AT THE NEXT SLIDE** until you have identified the error yourself



UNIVERSITÀ DI TORINO

The solution

- Go to the function retrieveAllCitiesData
- Remove the following three lines



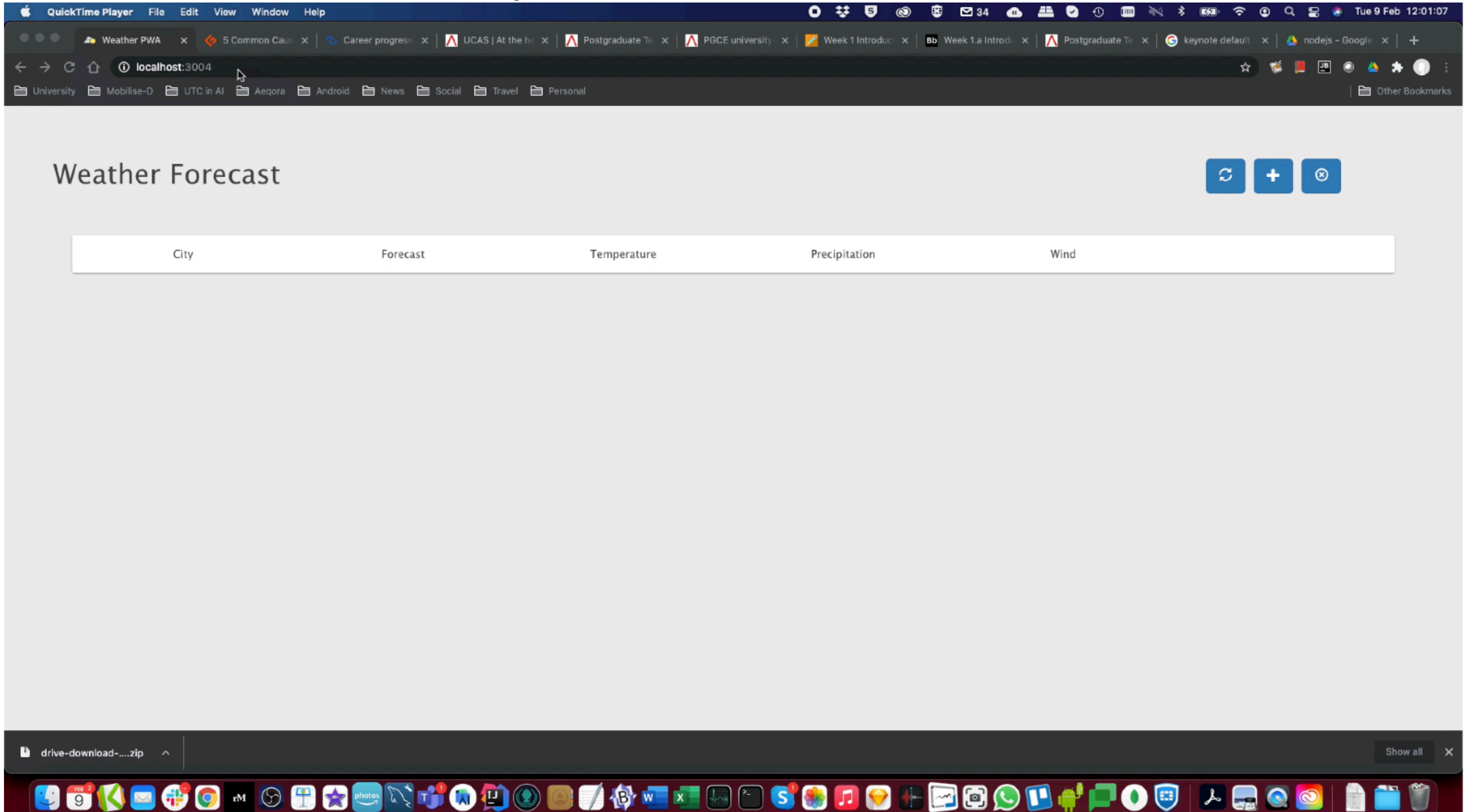
```
    let container=document.getElementById("container");
    container.innerHTML = "Common JS Bugs and Errors";
    if (!container) return;
```

- The error is due to the fact that there is not container element in the interface, so assigning its inner html causes an error

UNIVERSITÀ DI TORINO

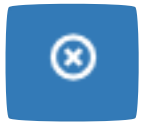
Video: how to debug the client side error

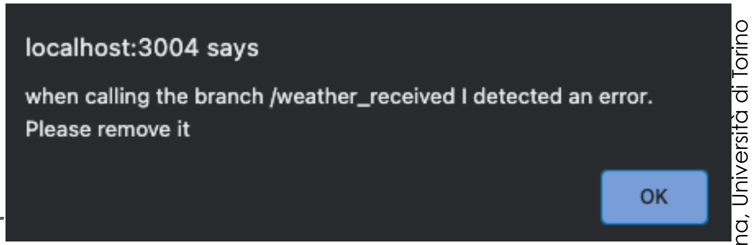
- To run the video, please check it out on e-Learn



The server-side error

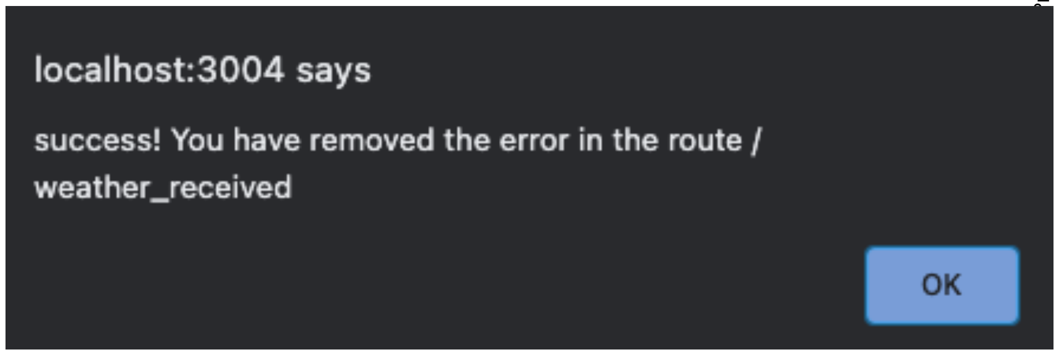
- When the client-side error is fixed, check the error on the server side
  - reload the page and you should see the correct cities and temperatures
  - click on the blue button with a cross in a circle (top right of screen)
    - an alert will tell you there is an error on the route /weather_received
  - identify the route */weather_received* in routes/index.js
  - find the error by putting a break on the first line and stepping until the error shows up
    - it may either happen that you will lose control of the debugger
    - or the entire server will crash
  - the server will return an error that is intercepted by an alert
  - remove the offending line
  - reload the server
  - press the button again
  - The alert should tell you you have removed the error





localhost:3004 says

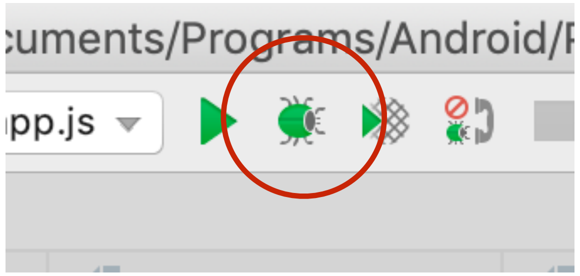
when calling the branch /weather_received I detected an error. Please remove it



localhost:3004 says

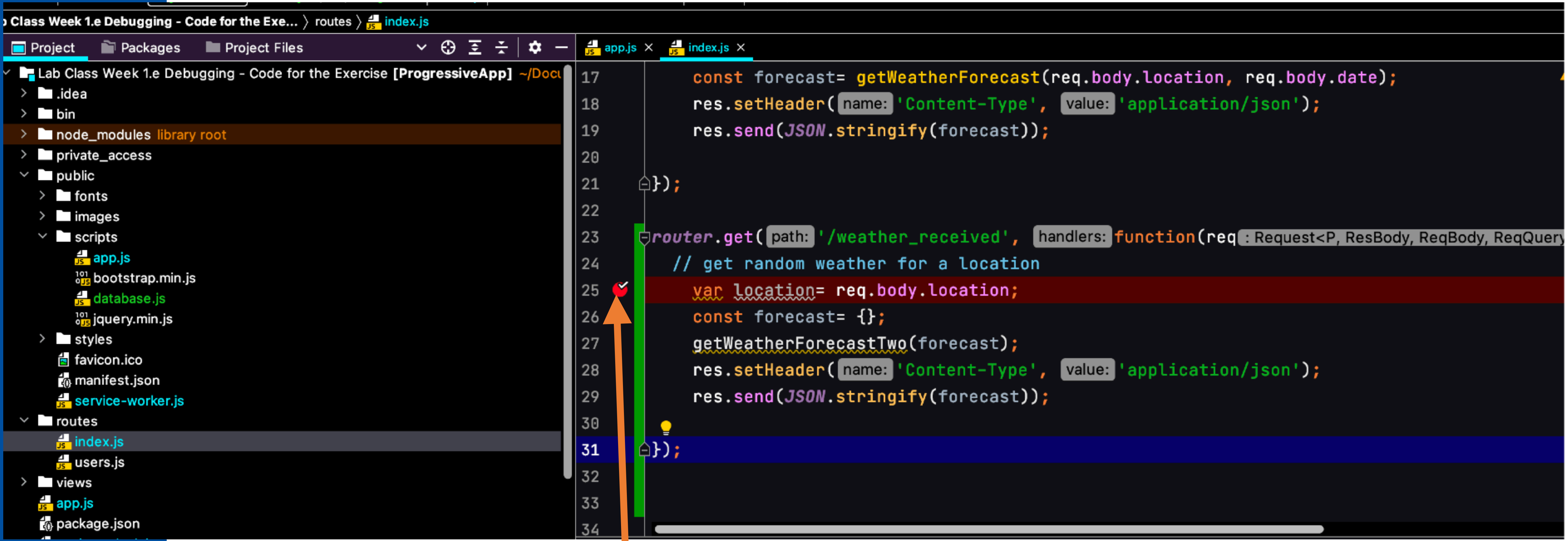
success! You have removed the error in the route / weather_received

How to debug the Server



- Click on the green bug
- identify the error in WebStorm's console
  - it will tell you that there is a 500 error on a specific route (which is the route that is called when clicking the button)
- Put a break point on the first line of the route that causes the error
- call the debugger
- understand why there is an error

How to set a break point



Class Week 1.e Debugging - Code for the Exe... 〉 routes 〉 index.js

```
Lab Class Week 1.e Debugging - Code for the Exercise [ProgressiveApp] ~/Docu
  .idea
  bin
  node_modules library root
  private_access
  public
    fonts
    images
    scripts
      app.js
      bootstrap.min.js
      database.js
      jquery.min.js
    styles
    favicon.ico
    manifest.json
    service-worker.js
  routes
    index.js
    users.js
  views
  app.js
  package.json
```

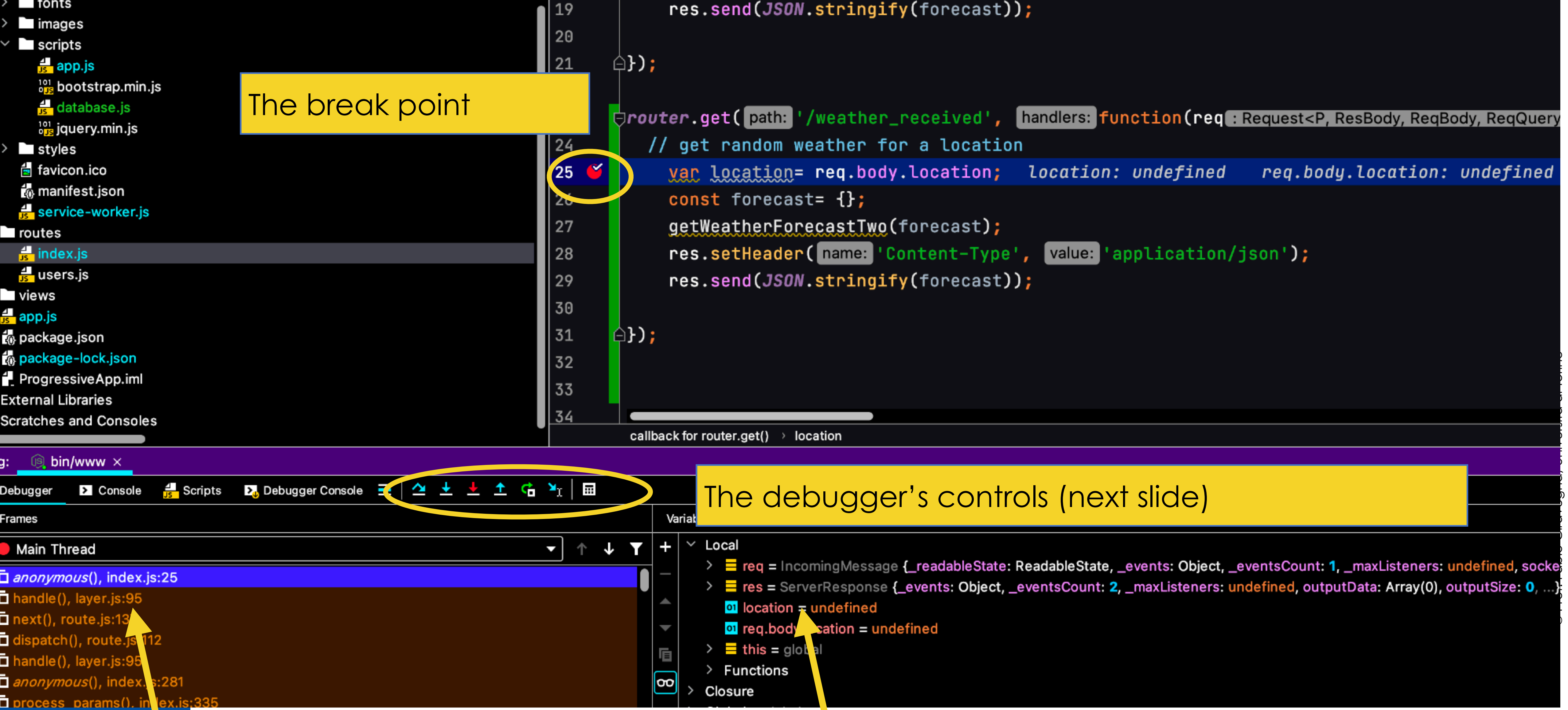
```javascript
    const forecast= getWeatherForecast(req.body.location, req.body.date);
    res.setHeader( name: 'Content-Type',  value: 'application/json');
    res.send(JSON.stringify(forecast));

});

router.get( path: '/weather_received',  handlers: function(req : Request<P, ResBody, ReqBody, ReqQuery
  // get random weather for a location
    var location= req.body.location;
    const forecast= {};
    getWeatherForecastTwo(forecast);
    res.setHeader( name: 'Content-Type',  value: 'application/json');
    res.send(JSON.stringify(forecast));

});
```

Click here

UNIVERSITÀ
DI TORINO

The debugger

- When the debugger stops at a break point you will see the debugger controls (next slide)



```
fonts
images
scripts
  app.js
  bootstrap.min.js
  database.js
  jquery.min.js
styles
favicon.ico
manifest.json
service-worker.js
routes
  index.js
  users.js
views
app.js
package.json
package-lock.json
ProgressiveApp.iml
External Libraries
Scratches and Consoles
```

```javascript
    res.send(JSON.stringify(forecast));

});

router.get( path: '/weather_received',  handlers: function(req : Request<P, ResBody, ReqBody, ReqQuery
  // get random weather for a location
    var location= req.body.location;   location: undefined    req.body.location: undefined
    const forecast= {};
    getWeatherForecastTwo(forecast);
    res.setHeader( name: 'Content-Type',  value: 'application/json');
    res.send(JSON.stringify(forecast));

});
```

callback for router.get() › location

```
anonymous(), index.js:25
handle(), layer.js:95
next(), route.js:13
dispatch(), route.js:112
handle(), layer.js:95
anonymous(), index.js:281
process_params(), index.js:335
```

```
Local
  > req = IncomingMessage {_readableState: ReadableState, _events: Object, _eventsCount: 1, _maxListeners: undefined, socke
  > res = ServerResponse {_events: Object, _eventsCount: 2, _maxListeners: undefined, outputData: Array(0), outputSize: 0, …}
    location = undefined
    req.body.location = undefined
  > this = global
  > Functions
> Closure
```

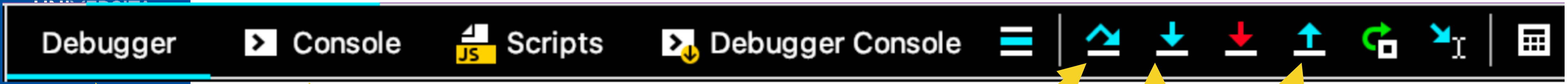
The break point

The debugger's controls (next slide)

The function call stack

The local and global variables' values

The debugger's controls



The debugger view (default view)

The javascript console (where you will spot errors; this is where the output of console.log() instructions goes)

The stepper - used to move the debugger execution to the next line. If the current line is a function call, it will execute its code as a one block. It will not step into the function's

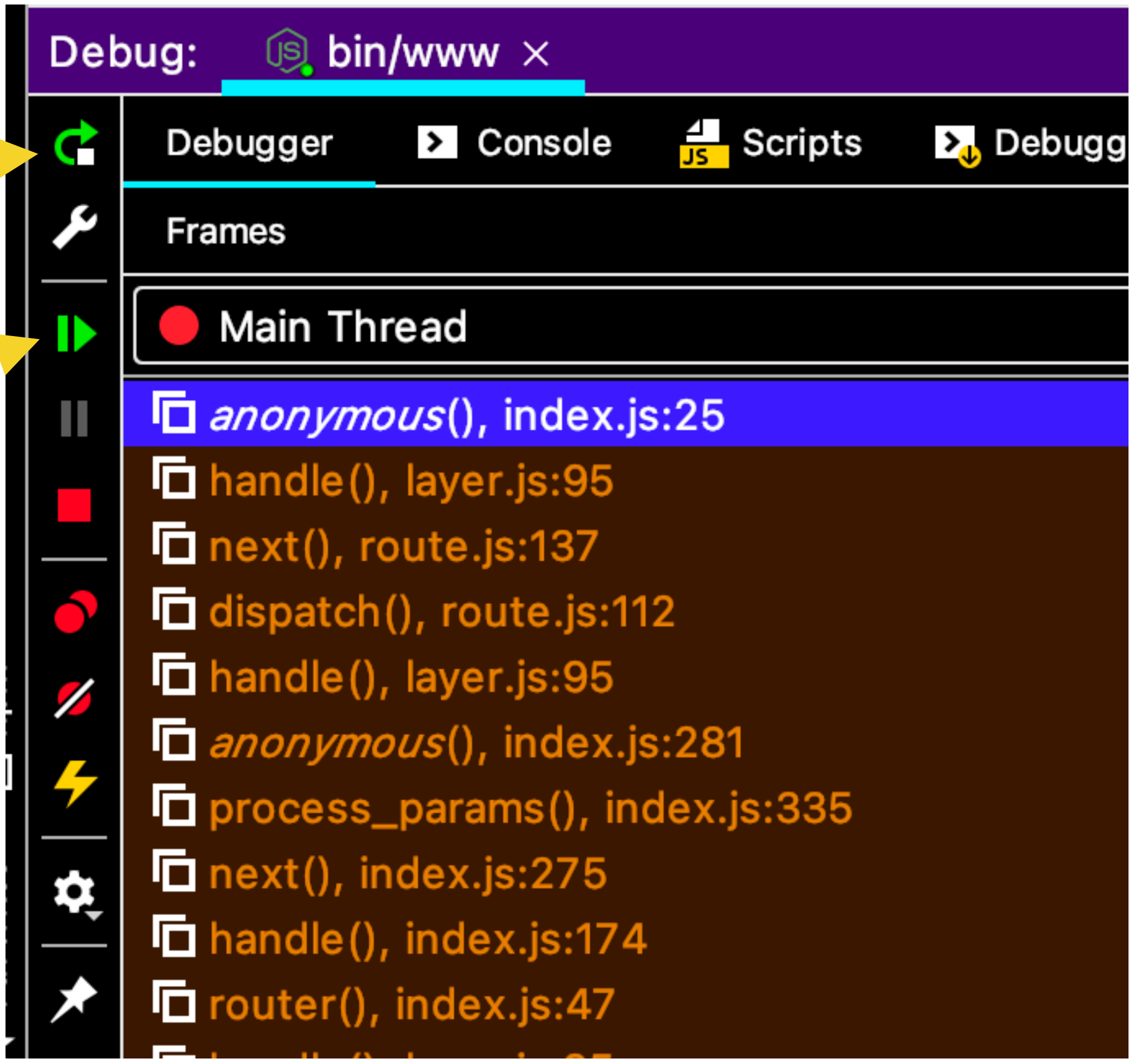
The stepper out a function - if you entered are debugging a function, it will step out of the function and continue debugging the calling function

The stepper into a function - if the current line contains a function call, use this button to step into the function's code

More controls (bottom left)

it runs the debugger from the start (equivalent to clicking on the green bug at the top of the

it moves the execution to the next break point. Useful when you are not interested in the current code and you want to proceed



anonymous(), index.js:25
handle(), layer.js:95
next(), route.js:137
dispatch(), route.js:112
handle(), layer.js:95
anonymous(), index.js:281
process_params(), index.js:335
next(), index.js:275
handle(), index.js:174
router(), index.js:47

Solution

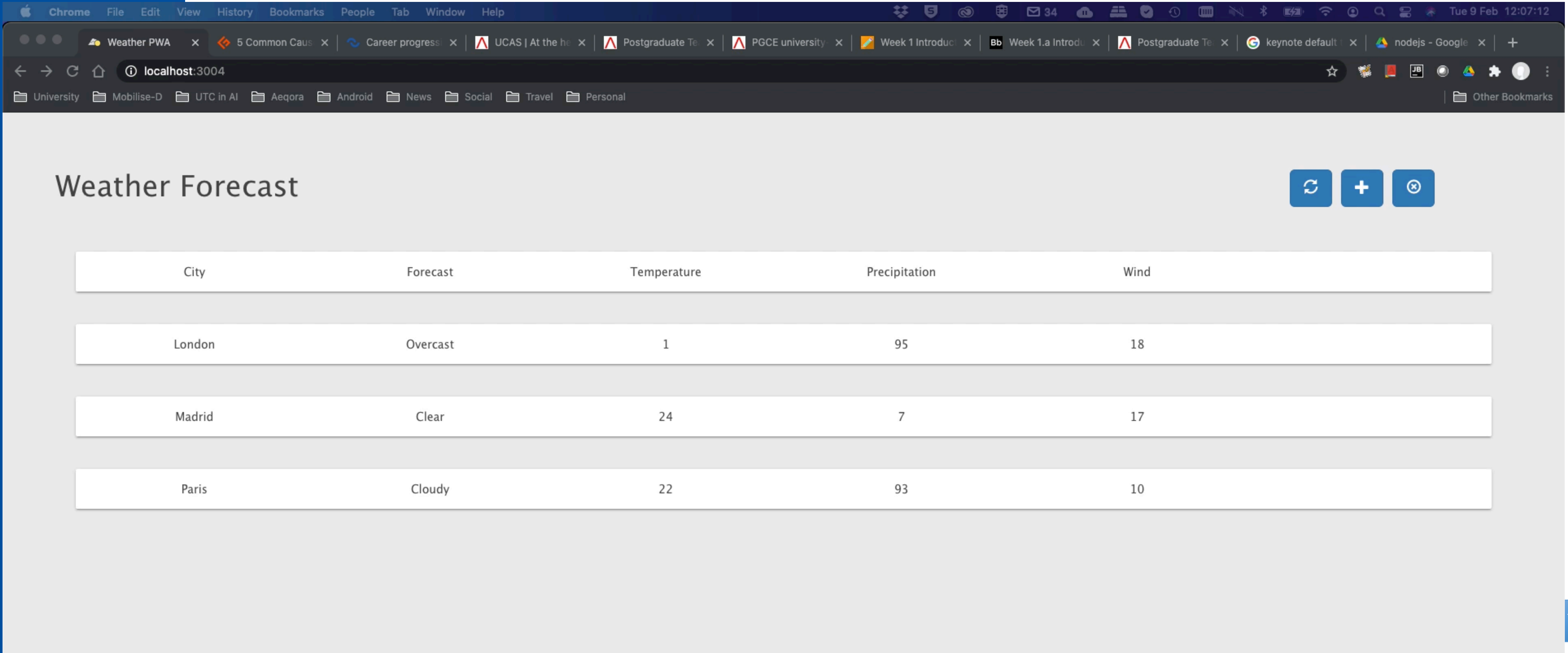
- comment out the following two lines

```
// var location= req.body.location;
const forecast= {};
// getWeatherForecastTwo(forecast);
```

- the run the server again
- it should work

Video: How to debug the server side error

- To see the video, go to the Solution folder


Chrome File Edit View History Bookmarks People Tab Window Help
Weather PWA
localhost:3004

Weather Forecast

City	Forecast	Temperature	Precipitation	Wind
London	Overcast	1	95	18
Madrid	Clear	24	7	17
Paris	Cloudy	22	93	10