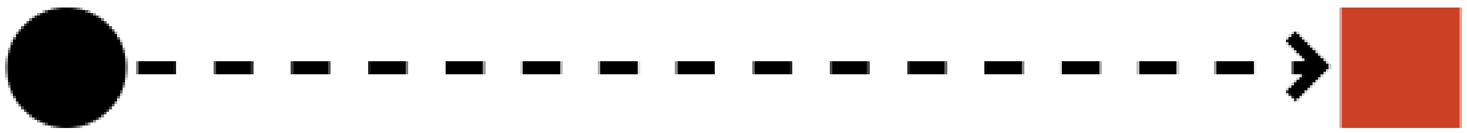
Transform data using RxJS

- Event Driven programming
  - had a save button, which exposes an onClick event, you would implement a confirmSave function, which when triggered, would show a popup to ask the user Are you sure?

onClick='confirmSave()'



Event Source

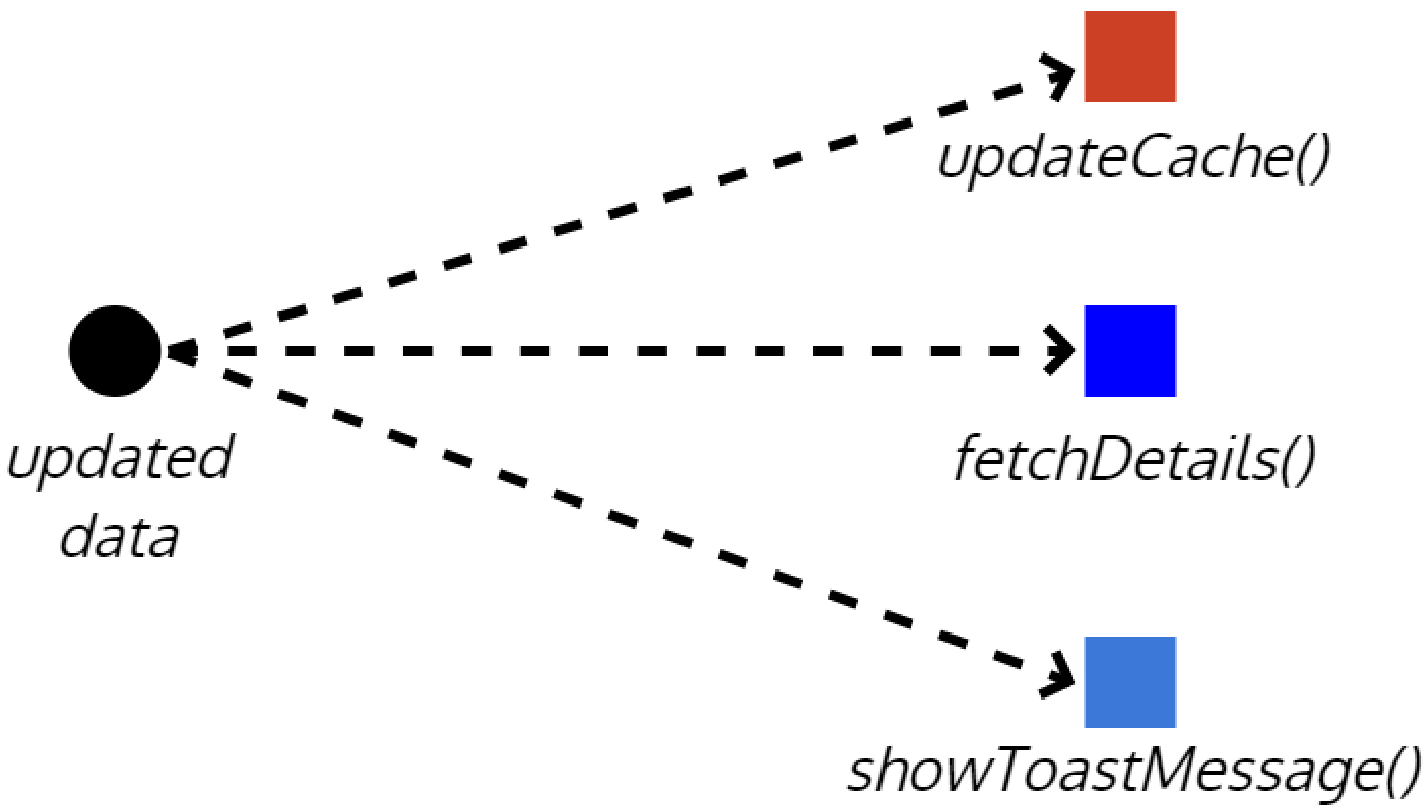
user clicks

Event Handler

window.alert('Are you sure?')

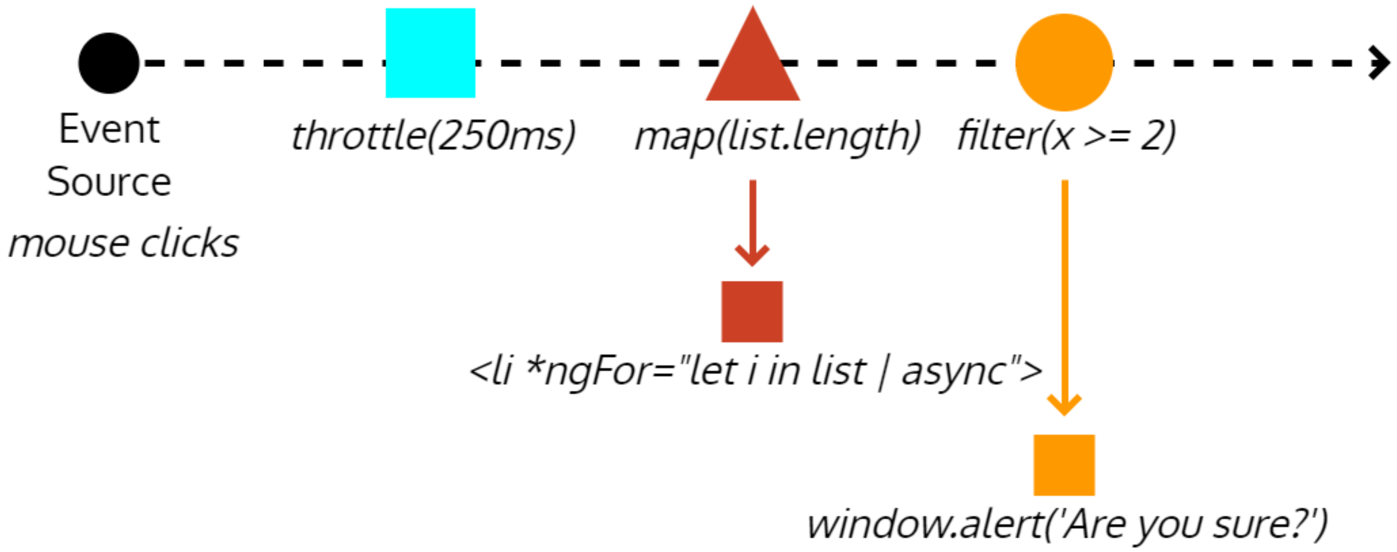
Event-Driven Implementation

- Publish subscribe pattern
- how can updated data event trigger an updateCache function that can update your local cache with new data, a fetchDetails function that can retrieve further details about the data from the server, and also a showToastMessage function that can inform the user that the app just received new data



Reactive Data Stream Implementation

- Pub/Sub Pattern Implementation
- Throttle 250ms, filter for double clicks, map for raw event data (# of clicks)
- display click data on an HTML list using *ngFor and Angular's async pipe, so the user can monitor the types of click data being captured every 250ms.



Implementing Reactive transformations

- To avoid future mistakes in returning the unintended type of data from your service, you need to update the getCurrentWeather function to define the return type to be Observable<ICurrentWeather> and import the Observable

```
src/app/weather/weather.service.ts
import { Observable } from 'rxjs'
import { ICurrentWeather } from '../interfaces'
...


export class WeatherService {
  ...
  getCurrentWeather(city:string, country:string): Observable<ICurrentWeather> {
  }
  ...
}
```

Reactive transformations

Now, VS Code will let you know that Type Observable<ICurrentWeatherData> is not assignable to type Observable<ICurrentWeather>:

1. Write a transformation function named transformToICurrentWeather that can convert ICurrentWeatherData to ICurrentWeather

2. Also, write a helper function named convertKelvinToFahrenheit that converts the API provided Kelvin temperature to Fahrenheit

```
src/app/weather/weather.service.ts

export class WeatherService {
...
  private transformToICurrentWeather(data: ICurrentWeatherData): ICurrentWeather {
    return {
      city: data.name,
      country: data.sys.country,
      date: data.dt * 1000,
      image: `http://openweathermap.org/img/w/${data.weather[0].icon}.png`,
      temperature: this.convertKelvinToFahrenheit(data.main.temp),
      description: data.weather[0].description
    }
  }

  private convertKelvinToFahrenheit(kelvin: number): number {
    return kelvin * 9 / 5 - 459.67
  }
}
```

Implementing Reactive Transformations

3. Update ICurrentWeather.date to the number type
4. Import the RxJS map operator right below the other import statements:

```
src/app/weather/weather.service.ts
import { map } from 'rxjs/operators'Current image
```

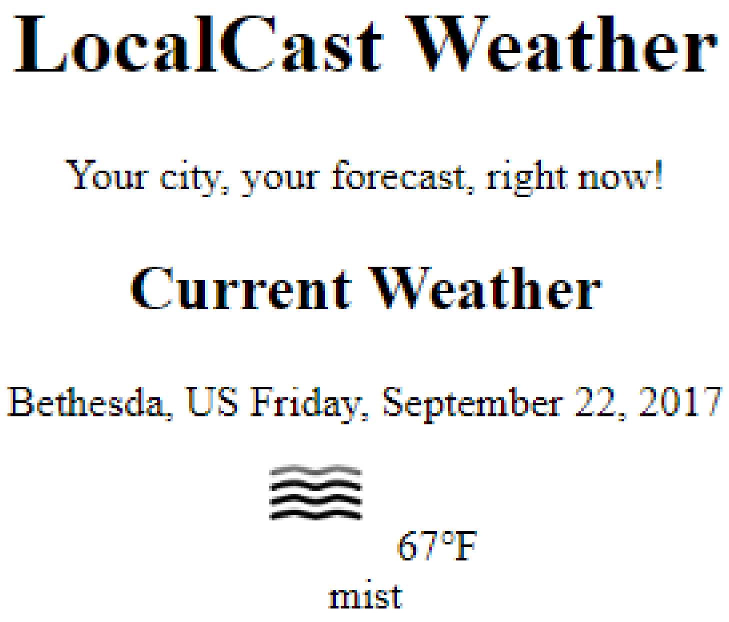
5. Apply the map function to data stream returned by httpClient.get method through a pipe
6. Pass the data object into the transformToICurrentWeather function:

```
src/app/weather/weather.service.ts
...
return this.httpClient
  .get<ICurrentWeatherData>(
    `http://api.openweathermap.org/data/2.5/weather?q=${city},${country}&appid=${environment.appId}`
  ).pipe(
    map(data =>
      this.transformToICurrentWeather(data)
    )
  )
...
```

Implementing Reactive Transformations

7. Ensure that your app compiles successfully

8. Inspect the results in the browser:



LocalCast Weather

Your city, your forecast, right now!

Current Weather

Bethesda, US Friday, September 22, 2017

67°F

mist