Rich Web

C15402002-wks-3

Lecture 7 review questions

Q1.

A stream is an abstraction of a sequence of data collected over some time. The sequence can be manipulated in many different ways. Stream allows this data to be processed chunk by chunk as the data appears in memory. Because streams are very sequential they can be processed very similarly to arrays and lists. But an array contains data that you already know and the size of it.

Streams can be created over anything including variables, user inputs and so on. Once a stream is created over some time the stream’s data can be operated on and transformed into new stream. This is known as the observer pattern. The data can be realised using the subscribe operation which can be processed using 3 stages: 1. Next: when new data items are available for processing, 2. Error: when an error has occurred during processing, 3. Complete: when there are no data left to process. Manipulation of stream can be very effective, it can be used to remove time components from codes which allows simplicity and is easy to debug.

Q2.

First using fromEvent to read input of mouse click or keyboard input implement an subscribe observer to listen continuously for new inputs. Using RXJS library request stream we can get an URL which will return a value fromPromise then use this as an observable. Using the function toMap we can get the value of the promise to a string. The data will then be rendered using subscribe.

Promises are often used to tackle problems with callbacks. Promise is a value that will resolve asynchronously like HTTP requests. Observables deal with sequence of asynchronous events. These events could be mouse positions, clicks, user input etc.  If the Observable object changes, it doesn’t necessarily have to require a re-write of anything else in the system. Multiple objects can register and observe the Observable object. Therefore, observables are richer and more complex abstraction for handling asynchronous requests. But, the downside is that observables cannot be chained. Observables return a subscription to the observable object. However, if tasks need to be chained promises can be chained together.