

Scala Concurrency

Practical Exercises

Chapter 1: Basic Concurrency with Scala

1. Define a function with the following signature

```
def getInt ( scale: Int ): Int
```

The function should return a randomly generated **Int** value between 0 and **scale**, after a random length delay. Invoke the function from within a future block, and verify that the behaviour is correct.

2. Now invoke the function twice, inside two different future blocks, and use a for comprehension to display the sum of the two results.
3. Modify the application to compare the two result values, returning true if the value of the f1 future is greater than the f2 future's value.
4. Modify the code of the function so that under certain circumstances it throws an exception. Ensure that the callbacks on your future are being invoked correctly. Verify also that the "post processing" you defined in earlier questions still operates successfully in the event a future failing to complete.