

⚠️ This course has been archived (Saturday, 17 December 2022, 00:00).

Course

🏠 CS-E4110

📖 Course materials

📊 Your points

📖 MyCourses 🔗

✉ Zulip Chat 🔗



**This course has already ended.**  
**The latest instance of the course can be found at: [Concurrent Programming: 2023](#)**

« [2 Roundrobin -- Creating children and forwarding messages.](#) [Course materials](#) [4 Supervisor -- Understanding supervisor strategies in Akka.](#) »

[CS-E4110](#) / [Round 4 - Concurrency frameworks](#) / 3 Actors Lifecycle -- Understanding lifecycle of Akka actors.

Assignment description

My submissions (1/10) ▾

## Actors Lifecycle -- Understanding lifecycle of Akka actors.

### Lifecycle

When you start creating more complicated Akka actors and actor hierarchies, you need to understand how the lifecycle of actors work. An actor's lifecycle is defined and can be controlled by the following methods: `receive`, `preStart`, `postStop`, `preRestart`, `postRestart`

### Code

Download the assignment template [here](#)

### Task

In this exercise, we implement an actor that helps to understand the lifecycle of an Akka actor. The overall goal of the task is to alter the lifecycle methods of an actor such that it prints the word 'SCALA' when we initialize it. The letters are sent from the provided Other.scala file. Look at the comments in the LifeCycle.scala file for further help.

### Note

Akka Classic actor APIs are used instead of the newer Akka Typed actor APIs in the exercise.

### Hint

You can find more information in [here](#) and [here](#).

📄 **LifeCycler.scala**

Choose File

No file chosen

Submit

Earned points

25 / 25



#### Exercise info

**Assignment category**  
Programming exercises

**Your submissions**  
1 / 10

**Deadline**  
Wednesday, 1 December 2021, 14:00

**Late submission deadline**  
Wednesday, 8 December 2021, 14:00 (-30%)

**Total number of submitters**  
47

« [2 Roundrobin -- Creating children and forwarding messages.](#) [Course materials](#) [4 Supervisor -- Understanding supervisor strategies in Akka.](#) »