

## Session #1: Introduction into the developer life

### Topic:

In this session you'll get an introduction into what it's like to be a professional developer working in a company. You'll work with several company developers, that will be asking you questions on what you think it's like to work in a company.

### What to focus on:

- How are these people different from you?
- How did they become a developer?
- How important is a Computer Science background?

### Write down 2 insights you've gained listening to the developers' personal stories:

1. Academic education isn't a must to get started and dig deeper in the programming world. Our mentor comes from Italy and he started doing programming as early as he was only 18 because he finds that interesting. He didn't even finish his college education, but through the time he never quitted learning hard by himself. He said most of the time you learn by yourself and through working, which also means personal motivation matters a lot.
2. Experience counts. Our mentor has worked in startup companies as well as big corporates. His varied experience is valued by the big company since he can bring more fresh insight. When you work in a small company, you will learn how to become an all-rounder.

### Summarize "company life as a professional developer" in your view (max. 150 words):

Company life most of the time means collaborate work which involves a lot of cooperation between different teams. What you do not only matters in the team, but also for the following project. Even a piece of code or an idea needs coordination. You need to make sure that it doesn't get in the way of other project while getting it work without bug. So communication and compromise is very important.

Also, you will never stop learning during company life. There's always problems that need to be fixed, codes that go obsolete. Always learning enables you to better detect problems and find solutions.

### Write down 2 soft skills you need to work on:

1. The ability to reflect on greater picture and to dig deeper into the concept. Our mentor mentioned that during a interview they would poke here and there to see the boundary of your knowledge, not only the coding skill but also your own opinion about the coding system itself that comes from a lot of practice and reflection. For example, what do you think is the defect of javascript. Most of the time, I just focus on the implementation rather than the deeper idea and concept.
2. Time managing and crisis control. I always have a lot of plans, but always fail to implement when things come rushing in. For example, it's hard to maintain the time/quality of learning or stay focused when I ran into some trouble in life. I should at least get the thing that I can do done first and then start worrying.