Training	A3.SW2 - Clean Coder / Clean Code

Keyfacts

Duration	2 months (2+1 days plus homework)
Language	English or German
Setting	On-site or remote

Target Group	Software Engineers

Training Goals	The ultimate goal of this training is to learn how (1) become a clean coder and (2) how to write clean code

#1 - Clean Coder
  Participants are aware of the role of a Clean Coder. They know about principles and practices and how to integrate them into their daily work as team to maintain a high level of software quality

#2 - Clean Code
  Participants understand the fundamental concepts of Clean Code and how to write and maintian Clean Code

Training Content	**Variation Points:** This training can be adopted to different programming languages. At present, the following languages are implemented and maintained: • Python • C# • C++ • C
	# Clean Coder • Clean Coding Basics • Clean Coder Behaviour and Habits • Clean Coding Principles & Practices • Mentoring & Craftmanship **# Clean Code** • Clean Coder & Clean Code • Clean Code Practices • Clean Code & Refactoring • Achieving Proficiency **# Practices** • Exercise: "Coding under Pressure" • Exercise: "Refactoring"

Learning Methods and Didactics	This training is designed as a 2 month learning and practicing journey comprising 2 theory days, homework, and 1 practice day.
Your Benefit	This training provides an introduction and overview on fundamental concepts of Clean Coder and Clean Code. It will help you to improve the quality of your daily output as individual and as team
Your Trainer	**FH-Prof. Dr. Christian Neureiter** neureiter@successfactory.cc

Christian is Professor at the School of Information Technology and Digitalisation at Salzburg University of Applied Sciences. As head of the "Center for Dependable Systems Engineering" he is an expert in this field and has profound knowledge on the matter.

Asides his academic role, Christian has 10+ years of experience as consultant and trainer

at the Successfactory Consulting group with a particular focus on Leadership, Software, and Systems Engineering related topics.

move to improve