

# Modulidentifikation



Modulnummer	320	
Titel	Objektorientiert Programmieren	
Kompetenz	Applikationen und Schnittstellen objektorientiert modellieren, implementieren, testen und dokumentieren.	
Handlungsziele	1	Analysiert Anwendungsprobleme zur Erstellung von objektorientierten Programmen. [g4.1, g4.4]
	2	Modelliert und dokumentiert objektorientierte Programme. [g4.4]
	3	Implementiert objektorientiertes Design. [g5.2, g5.5]
	4	Überprüft die Implementierung auf Korrektheit und Qualität. [g5.4, g6.3, g6.5, g6.6]
Kompetenzfeld	Application Engineering	
Objekt	Anwendung mit 3 bis 5 fachlichen Klassen.	
Modulversion	1.0	
Erstellt am	12.03.2021	

# Handlungsnotwendige Kenntnisse

Handlungsnotwendige Kenntnisse beschreiben Wissen, das die kompetente Ausführung der Handlungen eines Moduls unterstützt. Diese Kenntnisse dienen der Orientierung und sind nicht abschliessend definiert. Die daraus folgende Konkretisierung der Lernziele und das Festlegen des Lernwegs für den Kompetenzerwerb sind Sache der Bildungsanbieter.

Modulnummer	320	
Titel	Objektorientiert Programmieren	
Kompetenz	Applikationen und Schnittstellen objektorientiert modellieren, implementieren, testen und dokumentieren.	
Handlungsziele und handlungsnotwendige Kenntnisse		
1	1.1	Kennt den objektorientierten Ansatz mit den Grundkonzepten Kapselung, Vererbung und Polymorphie.
	1.2	Kennt Vorgehensprinzipien zur Klassenfindung (z.B. Spezialisierung und Generalisierung, Domänenmodell und –Events, BoundedContext, DRY, SRP)
	1.3	Kennt Abstraktionskonzepte wie Assoziationen, Klassen, Attribute und Methoden, Schnittstellen und abstrakte Klassen und Datentypen (ADT, Collections, Generics).
2	2.1	Kennt Beschreibungen (z.B. CRC-Card) und Diagrammnotationen (z.B. UML, TAM, 4+1) für statische und dynamische Aspekte von objektorientiertem Design.
	2.2	Kennt das Dokumentieren mit einer Dokumentationsinfrastruktur (z.B. Javadoc).
3	3.1	Kennt den Unterschied zwischen Klassen und Objekten.
	3.2	Kennt eine objektorientierte Sprache und deren Elemente und kann damit das Design umsetzen.
	3.3	Kennt das Konzept der dynamischen Bindung.
	3.4	Kennt das Konzept der Inversion of Control (DependencyInjection)
4	4.1	Kennt Verfahren zum Erstellen von Testfällen.
	4.2	Kennt Testmethoden zur Überprüfung der Komponenten einer Anwendung (automatisches Unit-Testing).
Modulversion	1.0	
Erstellt am	12.03.2021	