

LEHRKRAFTFASSUNG · BKWI2

Musterlösung und Erwartungshorizont

Name	Klasse	Datum	Punkte
			___ / 70

Bearbeitungszeit: 120 Minuten · Gesamtpunktzahl: 70 · Arbeitsfassung v1.2

Hilfsmittel: Entwicklungsumgebung und Structogrammer nach schulischer Festlegung. KI-Systeme nur, wenn die Lehrkraft dies ausdrücklich erlaubt.

Kompetenzzuordnung

Aufgabe	Lehrplanbezug	Kompetenz
1	BPE 8.1	Klassen, Interfaces, Assoziation, UML
2	BPE 8.1	Polymorphie und Überschreiben
3	BPE 8.2/8.3	MVC und Schichten
4	BPE 8.3	JDBC, CRUD, ResultSet, Container
5	BPE 8.3	SQL-Injection, Prepared Statement, Hashing/Verschlüsselung
6	BPE 8.1–8.3	Testing, Refactoring, Erweiterbarkeit

Musterlösung Aufgabe 1

Team aggregiert 0..* Mitglieder; Mitglied implementiert Beitragspflicht. Beitragspflicht definiert `double beitrags()`. Attribute privat, fachliche Methoden statt pauschaler Setter.

Musterlösung Aufgabe 2

`Beitragspflicht` als gemeinsamer Vertrag; zwei überschreibende Implementierungen; Aufruf über `List<Beitragspflicht>` ohne Typabfrage.

Musterlösung Aufgabe 3

View: Ein-/Ausgabe; Controller: Ereignisablauf; Service: Fachanwendungsfall/Validierung; Repository: Persistenz. Fehler wird fachlich oder technisch abgegrenzt und verständlich angezeigt.

Musterlösung Aufgabe 4

SQL mit `??`; Connection, PreparedStatement und ResultSet in try-with-resources; `setInt/setString`; Mapping aller Spalten; Optional.empty bei fehlender Zeile.

Musterlösung Aufgabe 5

Konkatenation erlaubt SQL-Injection. PreparedStatement trennt Befehl und Daten. Passwörter werden mit Salt gehasht; verschlüsselt werden Daten, deren Klartext später benötigt wird.

Musterlösung Aufgabe 6

Beispiele: Grenztest Beitrag, Repository-Fehlerfall, neue Beitragsstrategie ohne Clientänderung; Refactoring mit Tests vor und nach Extraktion.

Bewertungsraster

Bereich	Beobachtbare Kriterien	Punkte
Modell	Klassen/Interface 4 · Attribute/Methoden 4 · Multiplizitäten 3 · Kapselung 2 · Lebensdauerbegründung 2	15
Polymorphie	Vertrag 3 · zwei Implementierungen 5 · dynamischer Aufruf 3 · keine Typabfrage 2 · Test 1	14
Architektur	vier Verantwortungen 4 · Ablauf 4 · Fehlerweg 2	10
JDBC	SQL/Platzhalter 3 · Ressourcen 3 · Bindung 2 · Ausführung 2 · Mapping/Optional 4 · Fehlergrenze 2	16
Sicherheit	Risiko 2 · PreparedStatement 3 · Hashing 2 · Verschlüsselung 2 · Fälle 1	10
Qualität	zwei Tests 3 · Refactoring mit Nachweis 2	5

Notenschlüssel für 70 Punkte: ab 95 % sehr gut · ab 80 % gut · ab 65 % befriedigend · ab 50 % ausreichend · ab 30 % mangelhaft · darunter ungenügend. Schulisch verbindliche Vorgaben gehen vor.

Typische Fehler und Alternativen

Typische Fehler: Typabfrage statt Polymorphie, SQL in der View, unvollständiges Ressourcenmanagement, Parameterkonkatenation, Gleichsetzung von Hashing und Verschlüsselung. Gleichwertige DAO-/Repository-Bezeichnungen und andere fachlich konsistente Beitragssätze sind zulässig, sofern Vertrag und Prüfkriterien erfüllt sind.