

Datenbankendesign

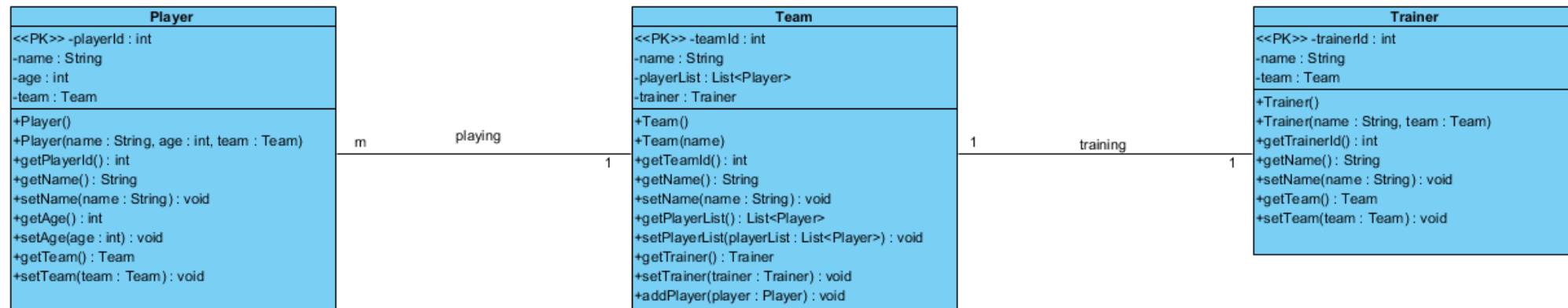
(ILV, SS 2019)

Dipl.-Ing. Muesluem Atas, BSc

7160 Institut für Softwaretechnologie,
7000 Fakultät für Informatik und Biomedizinische Technik,
TU Graz

Aufgabe 7 (I)

- Um Hibernate Framework üben zu können werden wir folgendes Beispiel implementieren:
 - Eine Fußball-Datenbank soll alle Informationen über die Fußballmannschaften, Trainer und Spieler persistent speichern.
 - Sie sollen zuerst alle Klassen wie im Klassendiagramm abgebildet in Java implementieren und dann mit Hilfe von Hibernate Framework die Java Klassen (Plain Old Java Objects) in einer relationalen Datenbank (MySQL) abspeichern.

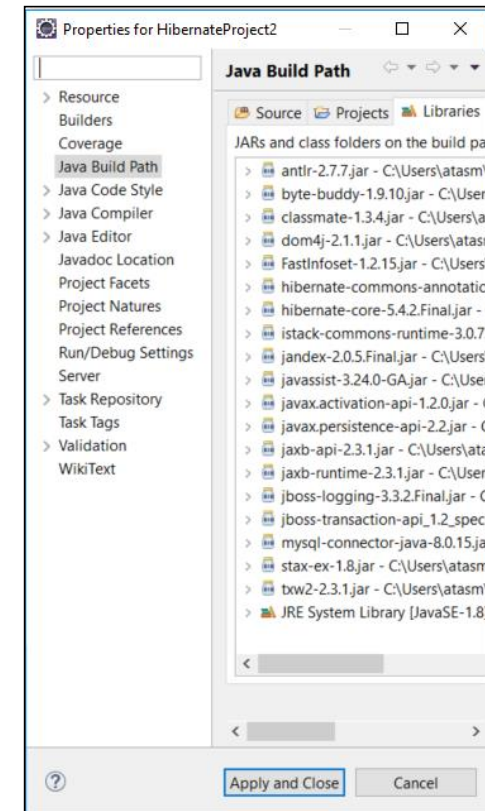


Aufgabe 7 (II)

- Erstellen Sie dafür zuerst die Konfigurationsdatei `hibernate.cfg.xml` und platzieren Sie diese Datei im Stammverzeichnis des Klassenpfades der Anwendung.
- Anschließend soll eine Datenbank namens „dbd_team“ manuell/händisch erstellt werden. Die Datenbank soll `CHARACTER SET=„utf8“` und `Collation=„utf8_unicode_ci“` haben.
 - Die „dbd_team“ DB entspricht unserer Fußball-Datenbank und soll alle Informationen über die Fußballmannschaften, Trainer und Spieler persistent speichern.

Aufgabe 7 (III)

- Damit Sie überhaupt mit Hibernate Framework arbeiten können, müssen Sie als erstes die Hibernate und MySQL Libraries herunterladen und dann die Libraries ins Projekt importieren.
 1. Laden Sie bitte MySQL-JDBC Treiber und Hibernate Libraries herunter
 - Die Libraries befinden sich im Moodle!
 2. Importieren Sie alle JAR Dateien (Libraries) im Java Projekt
 - Project Properties => Java Build Path => Libraries => Add External JARs...



Aufgabe 7 (IV)

- Als nächstes importieren Sie von Moodle die `HibernateSupport.java` Datei und platzieren Sie diese Datei im Klassenpfad der Anwendung.
- Die Basis für Hibernate Framework sind nun gelegt. Sie müssen im nächsten Schritt alle konkreten Klassen (`Team.java`, `Player.java` und `Trainer.java`) mit allen Variablen, Konstruktoren und getter/setter Methoden wie folgt implementieren:

Player
<<PK>> -playerId : int -name : String -age : int -team : Team
+Player() +Player(name : String, age : int, team : Team) +getPlayerId() : int +getName() : String +setName(name : String) : void +getAge() : int +setAge(age : int) : void +getTeam() : Team +setTeam(team : Team) : void

Team
<<PK>> -teamId : int -name : String -playerList : List<Player> -trainer : Trainer
+Team() +Team(name) +getTeamId() : int +getName() : String +setName(name : String) : void +getPlayerList() : List<Player> +setPlayerList(playerList : List<Player>) : void +getTrainer() : Trainer +setTrainer(trainer : Trainer) : void +addPlayer(player : Player) : void

Trainer
<<PK>> -trainerId : int -name : String -team : Team
+Trainer() +Trainer(name : String, team : Team) +getTrainerId() : int +getName() : String +setName(name : String) : void +getTeam() : Team +setTeam(team : Team) : void

Aufgabe 7 (V)

- Nachdem Sie alle konkreten Klassen mit Variablen, Methoden und Konstruktoren erstellt haben, sollten Sie diese Klassen mit Hibernate Annotations erweitern.
 - Bitte vergessen Sie nicht auch die Kardinalitäten der Beziehungen zwischen den Objekten anzugeben.
- Als nächstes erstellen Sie die Klasse `DatabaseConstruction.java`. Die Klasse soll die notwendigen Tabellen in der „dbd_team“ Datenbank automatisch erstellen.

Aufgabe 7 (VI)

- Nun erstellen Sie eine Schnittstelle „ISaveAndDelete.java“, die folgende Methoden beinhaltet:
 - `public boolean saveToDB();`
 - Speichert ein Objekt in der Datenbank.
 - `public void deleteFromDB();`
 - Entfernt ein Objekt aus der Datenbank.
- Folgende konkreten Klassen sollen die zuvor definierte Schnittstelle verwenden:
 - `Player.java`
 - `Team.java`
 - `Trainer.java`

Aufgabe 7 (VII)

- Zum Schluss erstellen Sie eine Klasse „TestDB.java“, welche folgende Spieler, Mannschaften und Trainer in der Datenbank erstellt.

player

player_id	age	name	team_team_id
1	29	Toni Kroos	3
2	29	Gareth Bale	3
3	26	Mohamed Salah	4
4	27	Sadio Mane	4
5	31	Lionel Messi	1
6	30	Sergio Agüero	2

team

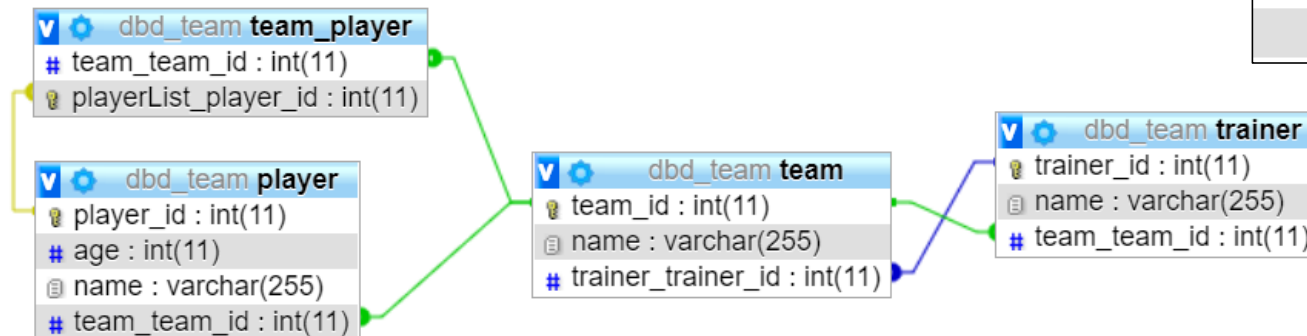
team_id	name	trainer_trainer_id
1	FC Barcelona	1
2	Manchester City	2
3	Real Madrid	3
4	FC Liverpool	4

trainer

trainer_id	name	team_team_id
1	Ernesto Valverde	1
2	Pep Guardiola	2
3	Zinedine Zidane	3
4	Jürgen Klopp	4

team_player

team_team_id	playerList_player_id
1	5
2	6
3	1
3	2
4	3
4	4



Aufgabe 7 (VIII)

- Deadline: 18.06.2019 23:59
- Abgabe: Geben Sie bitte die `ISaveAndDelete.java`, `DatabaseConstruction.java`, `Player.java`, `Team.java`, `Trainer.java`, `TestDB.java` und `Hibernate.cfg.xml` Dateien über Moodle ab.