

Inverted Exam

Angabe SSH-Race

Als Motorsportfan haben sie sich mit ihren Freunden eine Datenbank angelegt mit wichtigen Informationen über Fahrer, Strecken, Teams, Marken, Länder und Städte. Da Sie bereits Erfahrung mit Datenbanken haben, haben Sie sich bereiterklärt die Datenbank weiterzuführen und sind gebeten worden die folgenden Abfragen zu verbessern da sie nicht sehr performant sind und die Wartezeit eindeutig zu lang ist. Als weitere Schwierigkeit stellen Sie fest, dass der Code, wie so oft, wenn er von Hobby Entwicklern kommt, nur schlecht dokumentiert ist. Dies sollte für Sie aber kein Problem darstellen. Der Rückgabewert der einzelnen Methoden soll gleich bleiben um die Kompatibilität der API zu gewährleisten. Um Ihre Verbesserungen am Code zu testen, wurden JUnit-Tests bereitgestellt. Diese Tests dürfen nicht verändert werden. Sie sollen nach den Optimierungen in der angeführten Zeit durchlaufen.

Für besseres Verständnis hat Ihnen ihr Freund das ER-Diagramm auf der letzten Seite mitgegeben.

Vorbereitung

Um das System zum Laufen zu bringen befolgen sie bitte folgende Schritte:

- MariaDB(MySQL) muss gestartet sein
- „db.properties“ bearbeiten(Benutzer, Passwort, URL)
- Main.java ausführen um Datenbank zu erstellen und zu befüllen

Sollte beim Ausführen der ‚Main.java‘ die Meldung angezeigt werden „*jdbc.driver can't be loaded*“, stellen Sie sicher dass der entsprechende Datenbanktreiber aus dem ‚Vendor‘ Ordner im „Build Path“ Ihres Eclipseprojektes ist bzw. der eingetragene Treiber in „db.properties“ korrekt geschrieben ist.

Notenschlüssel

43-50 Punkte: Sehr Gut
35-42 Punkte: Gut
30-34 Punkte: Befriedigend
25-29 Punkte: Genügend
0-24 Punkte: Nicht Genügend

Name: _____

Aufgabe 1 – Heaviest Driver (10 Punkte)

Im Rennsport hat der Fahrer einen wesentlichen Einfluss. Dabei spielt der körperliche Zustand des Fahrers eine wesentliche Rolle. Deshalb ist es für ein Team nicht unwesentlich zu wissen in welcher körperlichen Verfassung der Fahrer ist. Um endlich die Diskussionen mit ihren Freunden zu beenden, soll nun der schwerste Fahrer ermittelt werden.

Allerdings ist die Abfrage welche den schwersten Fahrer zurückgibt nicht optimiert und braucht daher sehr lang bis die Ergebnisse vorliegen.

Es soll der zugegebenermaßen schlechte Algorithmus, der „*getHeaviestDriver*“ Methode aus der DriverDAO Klasse durch eine geschickte SQL-Abfrage ersetzt werden, die den schwersten Fahrer zurückliefert.

Derzeitige Laufzeit: ~7s
Zu erreichende Laufzeit: <0,5s
Erreichte Zeit: _____

Aufgabe 2 – Teams pro Land (10 Punkte)

Ein weiterer Diskussionspunkt ist, in welchem Land sich die meisten Teams niedergelassen haben. Die Meinungen Ihrer Freunde gehen auch hier auseinander und Sie wollen für das nächste Treffen gut vorbereitet sein und das Land mit den meisten Niederlassungen ausfindig machen. Die Abfrage lässt aber lange auf das Ergebnis warten.

Es soll wieder nur die Methode „*getCountryWithMostTeams*“ der CountryDAO Klasse bearbeitet werden. Wieder gilt es einen Algorithmus durch eine geschickte SQL-Abfrage auszutauschen.

Derzeitige Laufzeit: ~170s
Zu erreichende Laufzeit: <0,5s
Erreichte Zeit: _____

Aufgabe 3 – Fahrer pro Land (10 Punkte)

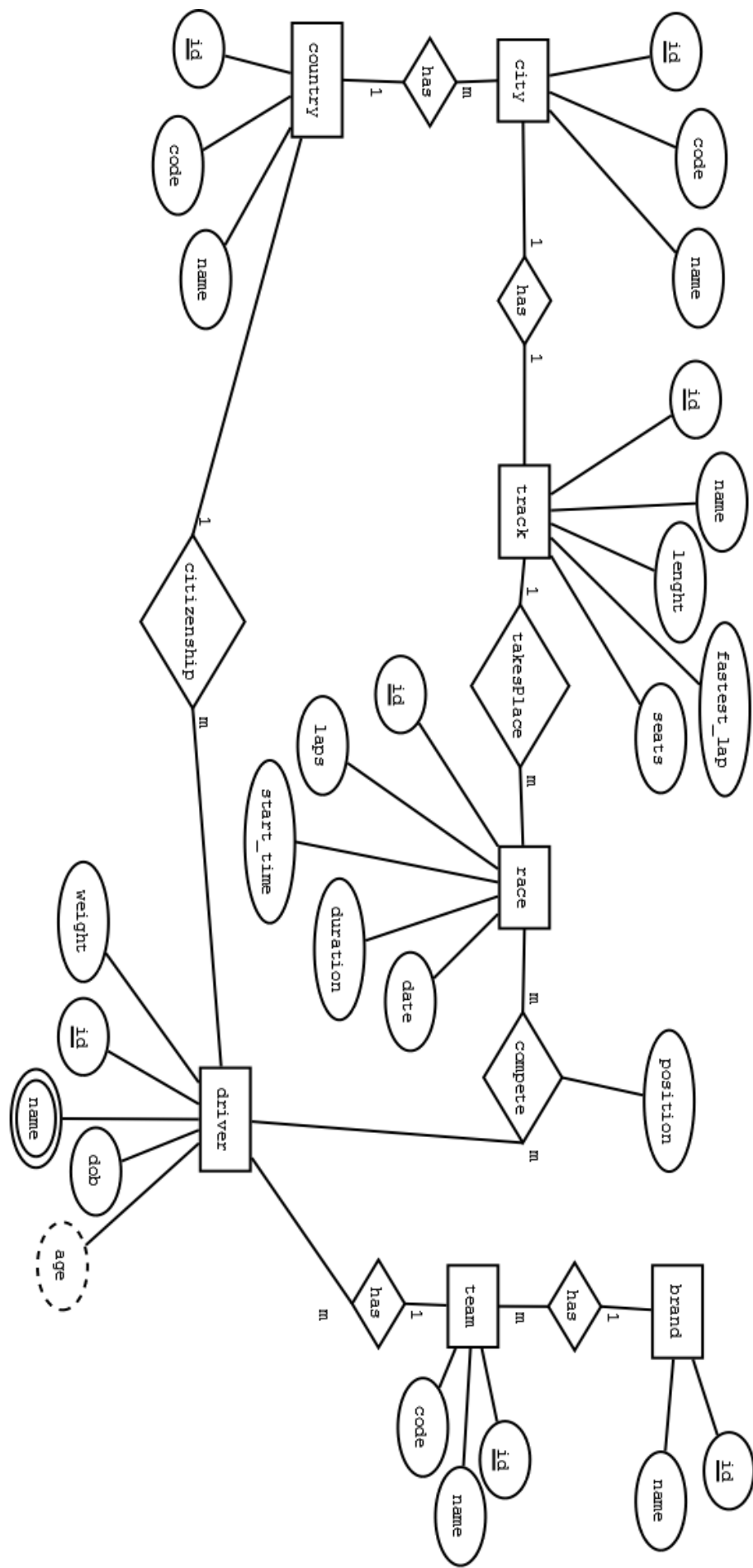
Jeder Motorsportfan will natürlich wissen wie viele Landsleute hinter dem Lenkrad platzen haben. Da die Abfrage aber momentan eine gefühlte Ewigkeit dauert, könnte diese Frage wohl an einem Abend nicht für jeden Ihrer Freunde geklärt werden. Es gilt also diese Abfrage zu optimieren.

Es soll abermals nur die Methode „*getDriversFromCountry*“ der DriverDAO Klasse verändert werden. Hier scheint die Abfrage nicht optimal für das gewünschte Ergebnis gewählt zu sein.

Derzeitige Laufzeit: ~5s
Zu erreichende Laufzeit: <0,5s
Erreichte Zeit: _____

VIEL ERFOLG!

Name: _____



Name: _____