

Übungen für die praktische Prüfung

1. Erstelle eine Datenbank, die Züge verwaltet und alle Informationen über die Routen der Züge enthält.
 - Die Datenbank enthält folgende Relationen (kann aber auch andere enthalten, **wenn nötig**): *Züge*, *Zugtypen*, *Bahnhöfe*, *Routen*
 - Jeder Zug hat einen Namen und gehört zu einem Zugtyp.
 - Der Zugtyp hat nur eine Beschreibung.
 - Jede Route hat einen Namen, einen zugeordneten Zug und eine Liste von Bahnhöfen mit dem Ankunftszeit und Abfahrtszeit des Zuges (z.B. der Zug kommt um 10:00 AM an und fährt um 10:10 AM ab)

Erstelle eine Datenbank in **Boyce-Codd Normalform** (mit den entsprechenden **Primär-** und **Fremdschlüsseln**), um die gegebenen Daten zu speichern, und füge ein paar Tupeln in die Relationen ein (sodass die Ergebnisse für die folgenden Aufgaben nicht die leere Menge sind)

2. Schreibe eine gespeicherte Prozedur, die Route, Bahnhof, Ankunftszeit und Abfahrtszeit als Parameter kriegt. Falls sich der Bahnhof schon auf dieser Route befindet, dann werden die Ankunftszeit und Abfahrtszeit entsprechend geändert. Ansonsten, wird der neue Bahnhof zu der Route eingefügt.
3. Schreibe eine benutzerdefinierte Funktion, die alle Bahnhöfe auflistet, die mehr als ein Zug haben an einem bestimmten Zeitpunkt. (z.B. um 10:30)
4. Erstelle einen Sicht mit den Namen der Routen, welche die wenigsten Bahnhöfe enthalten und, die, zusätzlich, nicht mehr als 5 Bahnhöfe enthalten.
5. Schreibe eine Abfrage, welche die Namen der Routen ausgibt, die alle Bahnhöfe enthalten.
6. Schreibe eine Abfrage, welche die Bahnhöfe ausgibt, wo ein Zug am längsten bleibt.
7. Erstelle einen Trigger um folgende Regel durchzusetzen. Wenn man einen neuen Bahnhof auf eine Route einfügt, dann muss man überprüfen, dass die Ankunfts- und Abfahrtszeiten nicht zwischen 3:00 AM und 5:00 AM sind. Falls das der Fall ist, dann wird eine Fehlermeldung angezeigt und die Daten werden nicht eingefügt.