

Testdokumentation

Aufgabe 1: CSVs einlesen

Fehler beim Encoding

Fehlerhafter Code:

```
BufferedReader br = new BufferedReader(  
    new InputStreamReader(getClass()  
        .getResourceAsStream(racesFileName)));
```

Da ich beim `InputStreamReader` kein UTF-8 Encoding angegeben habe, gab es in der Tabelle ein Problem mit dem Fahrer Räikkönen. In der Tabelle sah es so aus: `Kimi Räikkönen`

Verbesserter Code:

```
BufferedReader br = new BufferedReader(  
    new InputStreamReader(getClass()  
        .getResourceAsStream(racesFileName), "UTF-8")  
);
```

Beim `InputStreamReader` kann man nach der Ressource ein Encoding angeben. Dies hat den Fehler behoben.

In der Methode `readTeamsAndDriversFromFile()` wurde auch das Encoding vergessen und analog wie das obige Beispiel korrigiert.

Aufgabe 2: Import REST

Einlesen der Daten vom REST-Client:

Fehlerhafter Code - `readResultsFromEndpoint()`:

```
public void readResultsFromEndpoint() {  
    Response response = this.target  
        .request(MediaType.APPLICATION_JSON)  
        .get();  
    JsonArray payload = response.readEntity(JsonArray.class);  
  
    persistResult(payload);  
}
```

Code konnte nicht funktionieren, da `client` & `target` nie initialisiert wurden.

Verbesserter Code:

```
public void readResultsFromEndpoint() {

    this.client = ClientBuilder.newClient();
    this.target = client.target(RESULTS_ENDPOINT);

    Response response = this.target
        .request(MediaType.APPLICATION_JSON)
        .get();
    JsonArray payload = response.readEntity(JsonArray.class);

    persistResult(payload);
}
```

Fehlerhafter Code - `persistResult()`:

```
@Transactional
void persistResult(JsonArray resultsJson) {
    for (JsonValue jsonValue : resultsJson){
        Result result = new Result();
        result.setDriver(resultsJson.getJSONObject(0).getString("driverFullName")
    ); ①

        result.setPosition(resultsJson.getJSONObject(0).getInt("position")); ②
        result.setRace(resultsJson.getJSONObject().getInt("raceNo")); ③

        em.merge(result);
    }
}
```

① `SetDriver` cannot be applied to `java.lang.String` (Hätte überall Vettel eingetragen, da der Name vom 0. JSON Objekt Vettel ist)

② Hätte funktioniert, hätte aber jeden die Position 1 gegeben, da ich immer nur die Position vom 0. JSON Objekt eingetragen hätte

③ `SetRace` cannot be applied to `int`

Für `SetDriver` brauche ich ein `Driver` Objekt, bekomme aber einen `String`. Für `SetRace` brauche ich ein `Race` Objekt, bekomme aber einen `int`.

Verbesserter Code:

```

@Transactional
void persistResult(JsonArray resultsJson) {
    for (JsonValue jsonValue : resultsJson){
        JsonObject resultJson = jsonValue.asJsonObject();
        Result result = new Result();
        result.setDriver(em.createNamedQuery("Driver.findByName", Driver.class)
                                .setParameter("NAME", resultJson.getString(
"driverFullName")).getSingleResult());
        result.setPosition(resultJson.getInt("position"));
        result.setRace(em.createNamedQuery("Race.getById", Race.class)
                                .setParameter("ID", Long.valueOf(resultJson.
getInt("raceNo")))).getSingleResult());
        em.merge(result);
    }
}

```

Aufgabe 3: Gesamtpunkte eines Fahrers

Fehler 1: Java-Klasse RestConfig fehlte komplett.

Verbesserter Code:

```

@ApplicationPath("api")
public class RestConfig extends Application {
}

```

Fehler 2: `getPointsSumOfDriver()` wurde nicht gemacht

Verbesserter Code:

```

@GET @Produces(MediaType.APPLICATION_JSON)
public JsonObject getPointsSumOfDriver(@QueryParam("name") String name) {
    Long points;
    Driver driver;

    points = em.createNamedQuery("Result.pointSumDriver", Long.class)
        .setParameter("NAME", name)
        .getSingleResult();

    driver = em.createNamedQuery("Driver.findByName", Driver.class)
        .setParameter("NAME", name)
        .getSingleResult();

    return Json.createObjectBuilder()
        .add("driver", name)
        .add("points", points)
        .build();
}

```

Aufgabe 4: Sieger eines bestimmten Rennens

Fehler: wurde nicht gemacht

Verbesserter Code:

```

@GET @Path("winner/{country}") @Produces(MediaType.APPLICATION_JSON)
public Response findWinnerOfRace(@PathParam("country") String country) {

    Driver driver = em.createNamedQuery("Result.countryOfDriver", Driver.class)
        .setParameter("COUNTRY", country)
        .getSingleResult();

    return Response.ok(driver).build();
}

```

Aufgabe 5: Liste der Rennen, die ein Team gewonnen hat

Fehler: wurde nicht gemacht

Verbesseter Code:

```

@GET @Path("raceswon") @Produces(MediaType.APPLICATION_JSON)
public Response getWonRacesByTeam(@QueryParam("team") String team){

    List<Race> racesWon = em.createNamedQuery("Result.wonRacesTeam", Race.class)
        .setParameter("TEAM", team)
        .getResultList();
    return Response.ok(racesWon).build();
}

```

Aufgabe 6: Liste aller Fahrer mit ihren Punkten

Fehler: wurde nicht gemacht

Verbesseter Code:

```

@GET @Path("all") @Produces(MediaType.APPLICATION_JSON)
public Response getSumPointsOfAll(){
    List<Object[]> sumPointsDriver = em.createNamedQuery("Result.pointSumAll",
Object[].class).getResultList();
    return Response.ok(sumPointsDriver).build();
}

```