

Testdokumentation

Aufgabe 1:

- einlesen der Datei races.csv:

alter Code:

```
BufferedReader br = new BufferedReader(new
InputStreamReader(getClass().getResourceAsStream("/"+racesFileName),StandardCharsets.U
TF_8));

    while ((line = br.readLine()) != null) {
/* List <Race> races = this.em.createNamedQuery("Race.get",Race.class)
        .setParameter("id",row[0])
        .setParameter("country",row[1])
        .setParameter("date",row[2])
        .getResultList();*/

        Race race = new Race(row[0]);

        this.em.persist(race);
    }
```

neuer Code:

```
Race r = new Race(Long.parseLong(row[0]), row[1],
                    LocalDate.parse(row[2],
DateTimeFormatter.ofPattern("dd.MM.yyyy")));

em.persist(r);
```

Mit der Methode Long.parseLong wird der übergebene Wert (String) zu einem Long, für das einlesen des Datums muss außerdem DateTimeFormatter verwendet werden. Um auch Umlaute einlesen zu können: StandardCharsets.UTF_8

- einlesen der Datei teams.csv:

```
while ((line = bufferedReader.readLine()) != null) {
    String [] row = line.split(";");
    persistTeamAndDrivers(row);
}
```

Übergabe jeder Zeile an persistTeamAndDrivers als Array

Aufgabe 2:

falscher Code:

```
int raceNo = jsonValue.asJsonObject().getInt("raceNo");
```

richtig:

```
Long raceNo = Long.parseLong("" + jsonValue.asJsonObject().getInt("raceNo"));
```

Variable raceNo wurde als int und nicht als long gehandelt

falscher Code:

```
em.persist(new Result(em.find(Race.class, raceNo), pos, (Driver)
em.createNamedQuery("Driver.findByName", Driver.class).setParameter("NAME", name)));
```

richtig:

```
em.persist(new Result(em.find(Race.class, raceNo),
                        pos,
                        em.createNamedQuery("Driver.findByName", Driver.class)
                          .setParameter("NAME", name)
                          .getSingleResult()));
```

fehlte im alten Code: ".getSingleResult()"