

Advanced Topics in Databases

Practical Assignment

Aníbal Silva Rui Fernandes

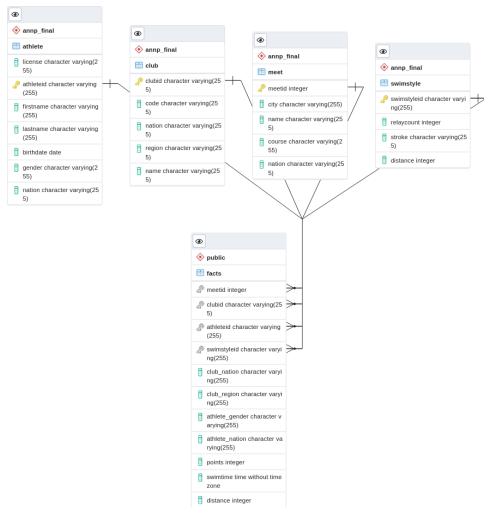
June, 2022

Database

Some changed parameters:

- ▶ `athleteid = firstname + lastname + birthdate + inc_id;`
- ▶ `clubid = code + nation + region;`
- ▶ `resultid = meetid + resultid;`
- ▶ `swimstyleid = distance + relaycount + stroke;`
- ▶ `eventid = meetid + eventid.`

Database



Average Age

```
SELECT AVG(age(birthdate))  
FROM annp_final.athlete;
```

- **Average Age:** 46 years, 6 months and 31 days.

Youngest Athlete

```
SELECT *  
FROM annp_final.athlete  
ORDER BY age(birthdate) ASC  
LIMIT 1;
```

- ▶ **Name:** Ana Mónica Eloi
- ▶ **Gender:** Female
- ▶ **Birthdate:** 29/12/1996
- ▶ **Age:** 25 years

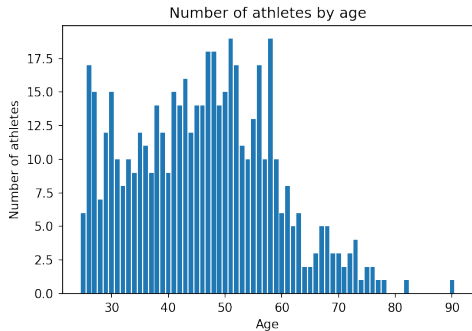
Oldest Athlete

```
SELECT *  
FROM annp_final.athlete  
ORDER BY age(birthdate) DESC  
LIMIT 1;
```

- ▶ **Name:** Virgílio Zacarias Costa
- ▶ **Gender:** Male
- ▶ **Birthdate:** 21/07/1931
- ▶ **Age:** 90 years

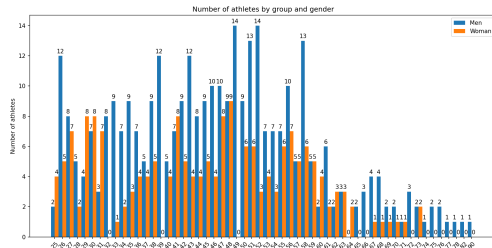
Athletes by Age

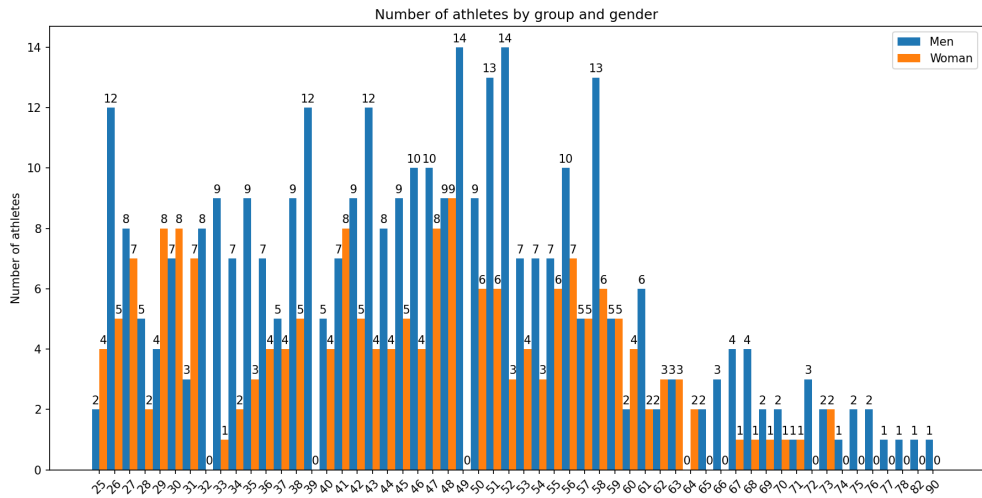
```
SELECT
    COUNT(*),
    EXTRACT(YEAR FROM age(birthdate)) AS age
FROM annp_final.athlete
GROUP BY age
ORDER BY age ASC;
```



Athletes by Age & Gender

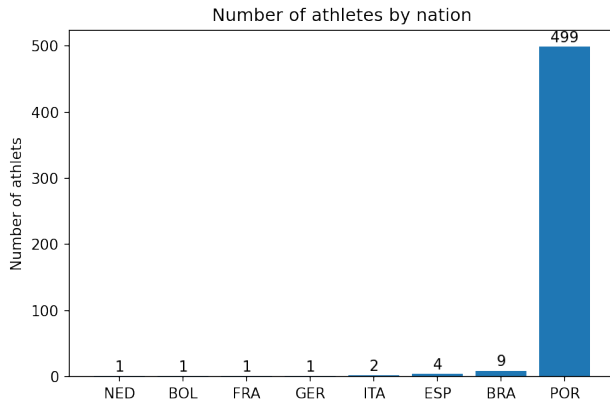
```
SELECT
    EXTRACT(YEAR FROM age(birthdate)) AS age,
    gender,
    COUNT(*)
FROM annp_final.athlete
GROUP BY (age, gender)
ORDER BY age ASC;
```





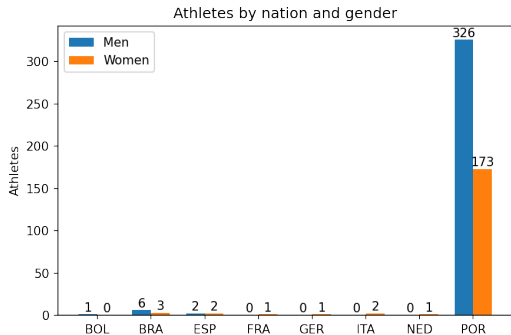
Athletes by Nation

```
SELECT
    nation,
    COUNT(*) AS nationCount
FROM annp_final.athlete
GROUP BY nation
ORDER BY nationCount ASC;
```

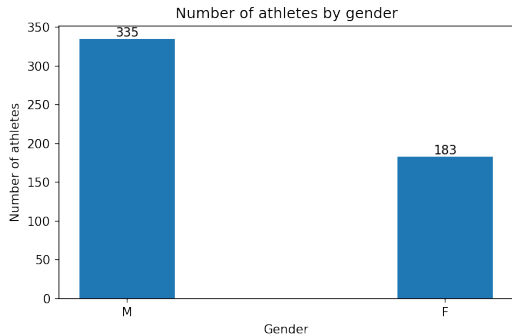


Athletes by Nation & Gender

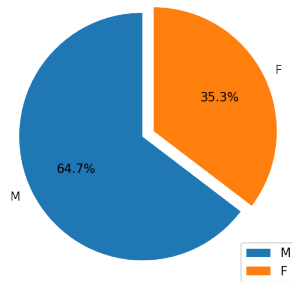
```
SELECT
    nation,
    gender,
    COUNT(*)
FROM annp_final.athlete
GROUP BY CUBE (nation, gender)
ORDER BY nation, gender NULLS LAST;
```



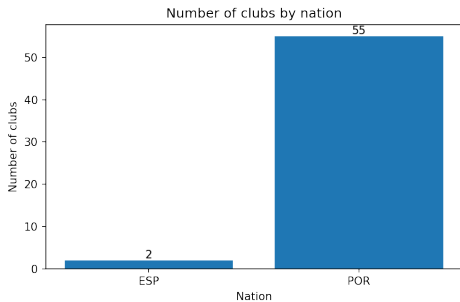
Athletes by Gender



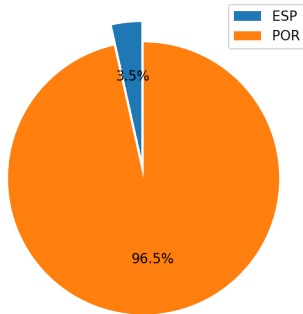
Percentage of athletes by gender



Clubs by Nation

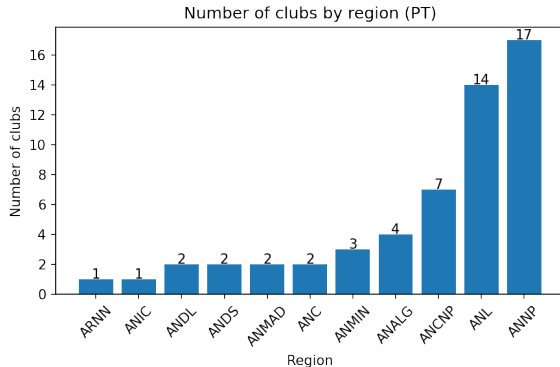


Percentage of clubs by nation

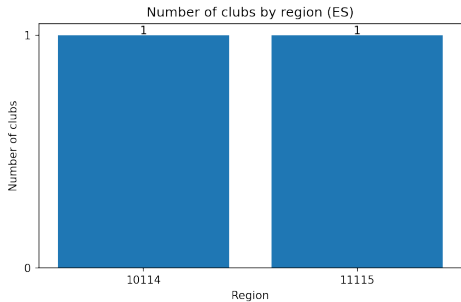


Clubs by Region (PT)

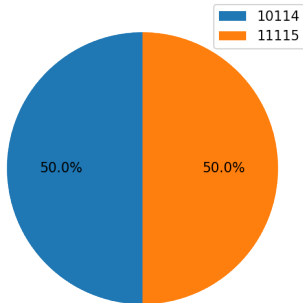
```
SELECT
    region,
    COUNT(*) AS regionCount
FROM annp_final.club
WHERE region SIMILAR TO '[A-Z]+'
GROUP BY region
ORDER BY regionCount ASC;
```



Clubs by Region (ES)



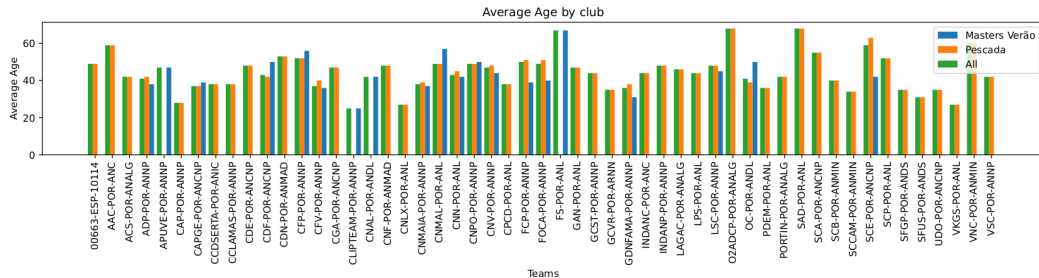
Number of clubs by region (ES)



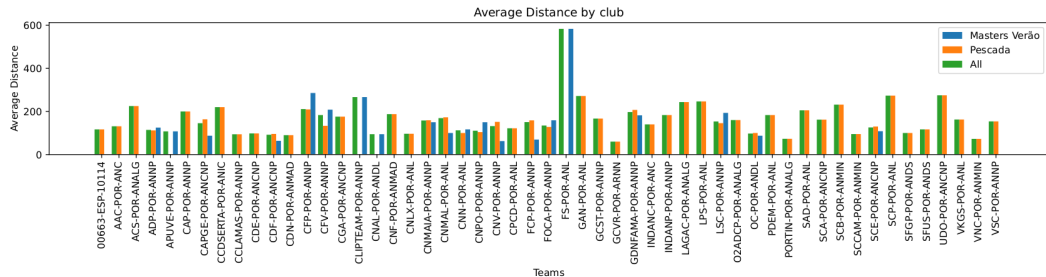
Club Fact statistics

```
SELECT
CASE GROUPING(af.clubid)
  WHEN 1 THEN 'all_clubs'
  ELSE af.clubid
END AS "Club",
CASE GROUPING(af.meetid)
  WHEN 1 THEN 'all_meets'
  ELSE af.meetid
END AS "Tournament",
ROUND(AVG(points), 2) AS "Average Points",
ROUND(AVG(distance), 2) AS "Average Distance",
ROUND(AVG(swimtime), 2) AS "Average Swimtime",
ROUND(AVG(birthdate)) AS "Average Age",
COUNT(af.athleteid) AS "Total Athletes"
FROM (
  SELECT
    CAST(meetid AS VARCHAR(255)),
    f.athleteid,
    clubid,
    points,
    distance,
    ROUND(CAST(
      (EXTRACT(
        MICROSECONDS FROM swimtime))/10^6
      AS NUMERIC), 2) AS swimtime,
    round(EXTRACT(
      EPOCH FROM
        CURRENT_TIMESTAMP - a.birthdate)/(365*24*60*60)) AS birthdate
  FROM facts f
  JOIN anmp_final.athlete a ON a.athleteid = f.athleteid
) af
WHERE points IS NOT NULL
GROUP BY CUBE (af.clubid, af.meetid)
ORDER BY "Average Points" DESC;
```

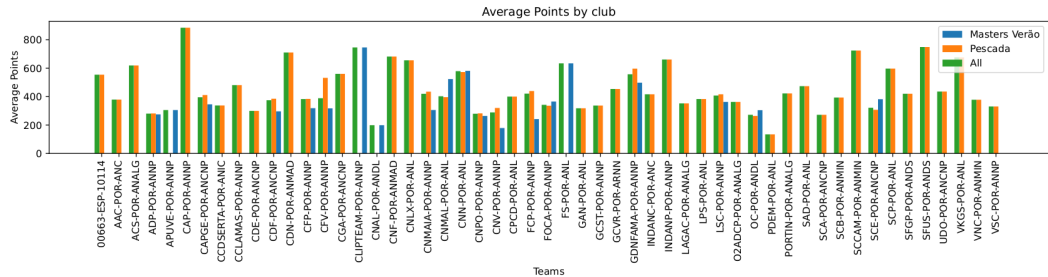

Overall Club Statistics I



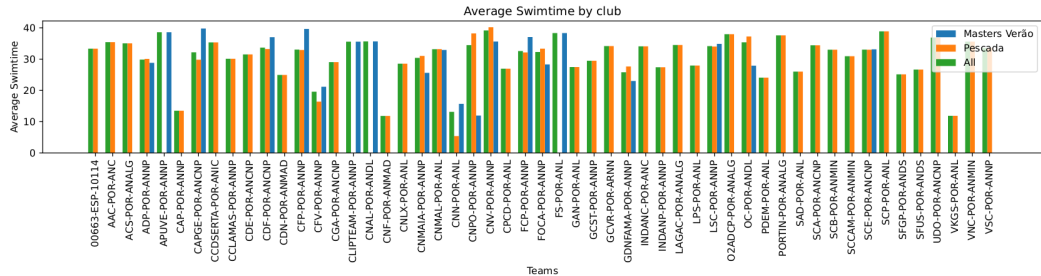
Overall Club Statistics II



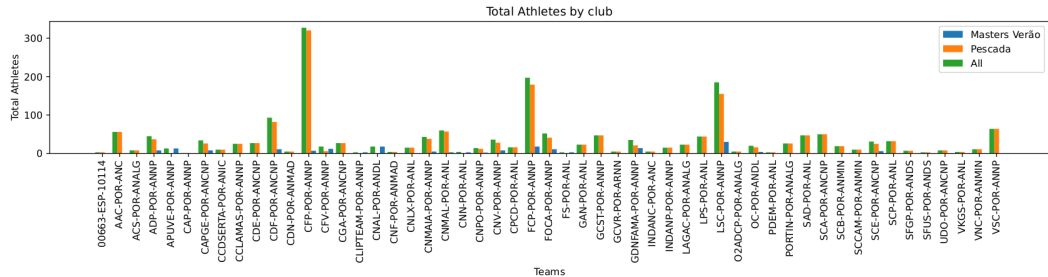
Overall Club Statistics III



Overall Club Statistics IV



Overall Club Statistics V



Swim Style Statistics By Club

```
SELECT
CASE GROUPING(a.firstname)
  WHEN 1 THEN 'all_players'
  ELSE a.firstname
END AS "Athletes",
CASE GROUPING(af.swimstyleid)
  WHEN 1 THEN 'All SwimStyles'
  ELSE af.swimstyleid
END AS "SwimStyle",
ROUND(AVG(points), 2) AS "Average Points",
ROUND(AVG(swimtime), 2) AS "Average Swimtime"
FROM (
  SELECT
    athleteid,
    CAST(swimstyleid as VARCHAR(255)),
    points,
    distance,
    ROUND(CAST(
      (EXTRACT(
        MICROSECONDS FROM swimtime))/10^6
      AS NUMERIC), 2) AS swimtime
    FROM facts) af
JOIN annp_final.athlete a ON a.athleteid = af.athleteid
WHERE points IS NOT NULL
GROUP BY CUBE (a.firstname, af.swimstyleid)
ORDER BY "Average Points" DESC;
```

Swim Style Statistics

