A logo of a book and tree

Description automatically generated

Advanced Topics in Databases

1st Deliverable

*Demertzoglou Efstratios | ΤΗ20580*

*Tatsios Ilias | TH20566*

Table of Contents

[The csv file I will be working with 3](#_Toc193767941)

[Functional dependencies: 3](#_Toc193767942)

[Initial Relational Schema 4](#_Toc193767943)

[Propose a Joinless Decomposition 5](#_Toc193767944)

[Candidate Key 6](#_Toc193767945)

# The csv file I will be working with

The Universal table is a group of attributes from the *races.csv*

UNIVERSAL(raceid,year,driverid,points,circuitid,alt)

# Functional dependencies:

raceid --> year;

raceid, points --> driverid;

raceid --> circuitid;

circuited --> alt;

# Initial Relational Schema

DROP TABLE IF EXISTS DRIVER\_STANDINGS CASCADE;

DROP TABLE IF EXISTS RACES CASCADE;

DROP TABLE IF EXISTS CIRCUITS CASCADE;

CREATE TABLE CIRCUITS(

circuitId INT NOT NULL,

alt INT NOT NULL,

PRIMARY KEY(circuitId));

COPY CIRCUITS FROM 'C:\uni\8x\ATD\archive\circuits\_modified.csv' DELIMITER ',' CSV HEADER;

CREATE TABLE RACES (

raceId INT NOT NULL,

circuitId INT NOT NULL,

FOREIGN KEY (circuitId) REFERENCES CIRCUITS (circuitId),

PRIMARY KEY (raceId));

COPY RACES FROM 'C:\uni\8x\ATD\archive\races\_modified.csv' DELIMITER ',' CSV HEADER;

CREATE TABLE DRIVER\_STANDINGS (

raceId INT NOT NULL,

driverId INT NOT NULL,

points INT NOT NULL,

FOREIGN KEY (raceId) REFERENCES RACES (raceId),

PRIMARY KEY(driverId,raceId));

COPY RESULTS FROM 'C:\uni\8x\ATD\archive\driver\_standings\_modified.csv' DELIMITER ',' CSV HEADER;

COPY (

    SELECT \* FROM CIRCUITS NATURAL JOIN RACES NATURAL JOIN RESULTS

) TO 'C:\uni\8x\ATD\archive\NATURALJOINTABLE.csv' DELIMITER ',' CSV HEADER;

By creating a natural join between these 3 tables as:

“*SELECT \* FROM CIRCUITS NATURAL JOIN RACES NATURAL JOIN DRIVER\_STANDINGS*”

The output projected should be the same to “*SELECT \* FROM UNIVERSAL*”

# Propose a Joinless Decomposition

A diagram of a circuit

AI-generated content may be incorrect.

# Candidate Key

UNIVERSAL(raceid,year,driverid,points,circuitid,alt)  
  
  
raceid --> year;

raceid, points --> driverid;

raceid --> circuitid;

circuited --> alt;

We have to prove that {raceid, points} is a candidate key  
  
X: desired candidate key = { raceid, points }  
  
1st step is to set X+ = { raceid, points }

F = { raceid → year }

F = { raceid → year , raceid 🡪circuit}

F = { raceid → year , raceid 🡪circuitid, circuitid 🡪alt}

F = { raceid → year , raceid 🡪circuitid, circuitid 🡪alt, raceid -> alt}

Therefore {raceid, points} is candidate key  
  
  
  
Link to all necessary CSVs and sql scripts:

<https://drive.google.com/drive/folders/1zmYyTNyI_6Miez2muLYqYLkN5V4PmFNU?usp=sharing>