University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 2

Date: March 3rd, 2025

Group Number: 6

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Nazif Ishrak	46844429	j9v7k	nzfishrak60@gmail.com
Edward Kim	97114250	c1p2i	eddiekim203@gmail.com
Daniel Owen Santosa	90451568	z1p5e	Daniel.o.santosa@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

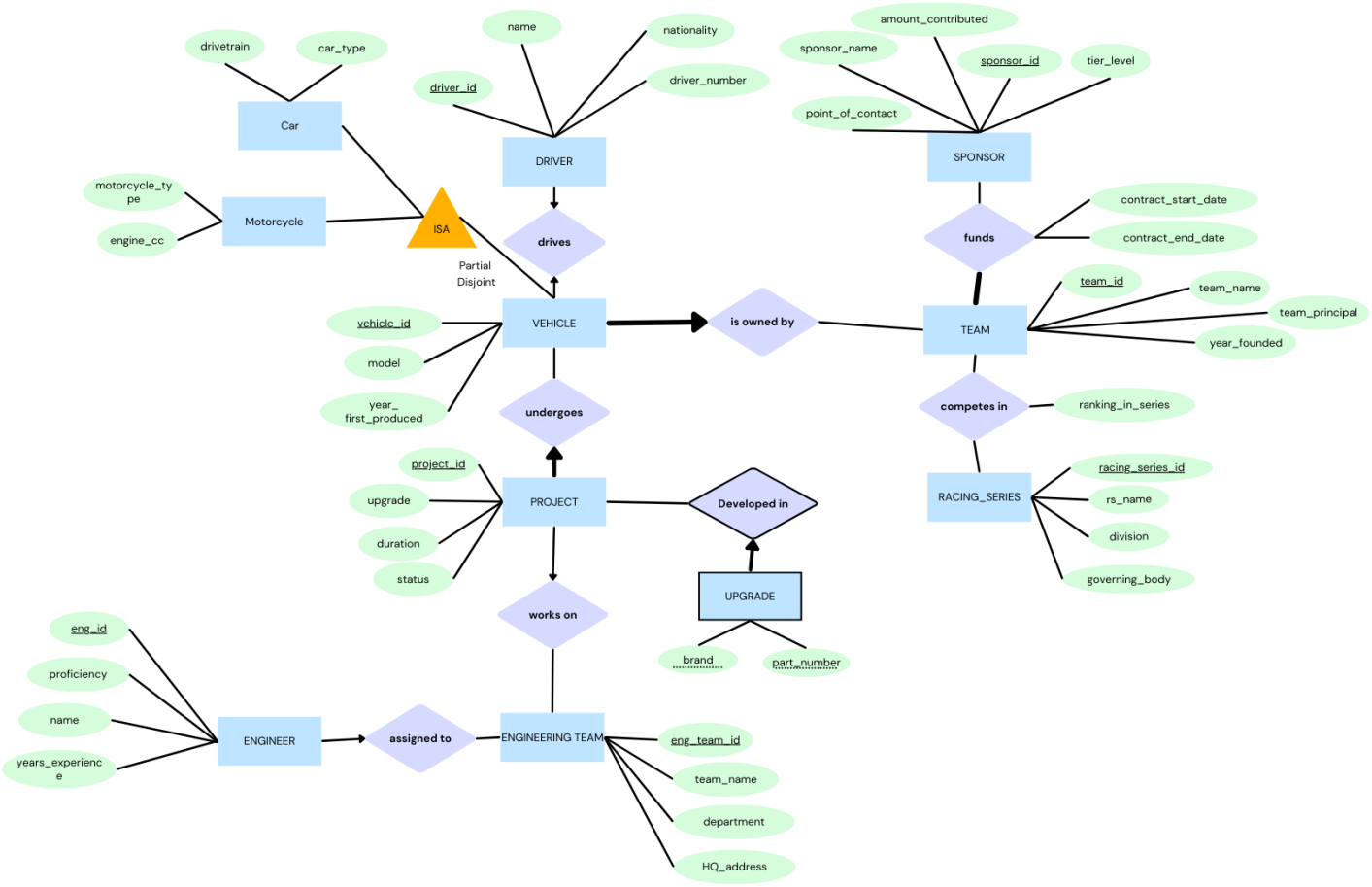
AI Declaration:

AI tools (ChatGPT) were used to improve wording of sentences and fix formatting of the document for this milestone.

University of British Columbia, Vancouver
Department of Computer Science

Summary & ER Diagram

This project is an application that enables efficient management of automotive racing teams and their engineering aspects. It does so by enabling systematic tracking of the implementation of various projects and upgrades in different vehicles, ensuring that the user will always know which team is on which project for which car. The application also allows teams to seamlessly manage these aspects across different categories of racing by being a dynamic database system.



University of British Columbia, Vancouver
Department of Computer Science

Changes Made to the ERD

DRIVER:

- Renamed d_id to driver_id for clarity
- Removed division attribute since it wouldn't make sense in our ERD. For that to occur, we'd need to have a relationship between Driver and Racing_Series. Since we've already gotten this far, it'd just take too much time anyway.
    - In our world, a driver can drive in different divisions
- Combined both fname and lname attributes to just name for simplicity.

VEHICLE:

- Renamed v_id to vehicle_id for clarity
- Renamed year_produced to year_first_produced for clarity
- Changed ISA relationship of Disjoint, Total to Partial, Total to also account for vehicles of different type not included in the ISA
- Removed Truck Entity of ISA relationship for simplicity (for now)

CAR (ISA of VEHICLE):

- Added drivetrain and car_type for additional attributes

MOTORCYCLE (ISA of Vehicle):

- Added engine_cc and motorcycle_type for additional attributes

PROJECT: No change

UPGRADE: No change

ENGINEERING_TEAM (was RESEARCH TEAM):

- Added HQ_address and department for additional attributes
- Changed team_id to eng_team_id for clarity

ENGINEER:

- Combined both fname and lname attributes to just name for simplicity
- Added proficiency and years_experience for additional attributes
- ***BIG CHANGE*** Removed ISA relationship with **Lead_Engineer** because we encountered a circular dependency with Research Team Works relationship and Lead Engineer Leads relationship.

University of British Columbia, Vancouver
Department of Computer Science

SPONSOR:

- No change, but we decided that sponsors (in our world) pay based on tier levels. This is so that we have functional dependencies, plus it simplifies our database (in the real world, contracts could be of varying amounts).
- Tier levels are as follows:
    - Bronze: $100,000
    - Silver: $500,000
    - Gold: $1,000,000
    - Platinum: $5,000,000
    - Diamond: $10,000,000

TEAM: No change

RACING_SERIES:

- racing_series_name is now rs_name and no longer a unique PK.
- Removed year as we felt tracking different seasons wasn't necessary
- racing_series_id is now the PK (this is to account for future name changes to the racing series)
- Added division to show the different forms of motorsport in life
- Added governing_body to add a bit of variety AND to show that different racing series is led by different groups

University of British Columbia, Vancouver
Department of Computer Science

Derived Schema

Sponsor(sponsor_id: INT, sponsor_name: VARCHAR(100), amount_contributed: INT, tier_level: VARCHAR(50), point_of_contact: VARCHAR(100))

Funds(**sponsor_id**: INT, **team_id**: INT, contract_start_date: DATE, contract_end_date: DATE)

Team(team_id: INT, team_name: VARCHAR(100), team_principal: VARCHAR(100), year_founded: CHAR(4))

Racing_series(racing_series_id: INT, rs_name: VARCHAR(100), division: VARCHAR(100), governing_body: VARCHAR(100))

Competes_In(**team_id**: INT, **racing_series_id**: INT, ranking_in_series: INT)

Vehicle(vehicle_id: INT, model: VARCHAR(100), year_first_produced: INT, **team_id**: INT, **driver_id**: INT)

Driver(driver_id: INT, name: VARCHAR(50), nationality: VARCHAR(50), driver_number: INT)

Project(project_id: INT, upgrade: VARCHAR(50), duration: INT, status: VARCHAR(50), **vehicle_id**: INT, **eng_team_id**: VARCHAR(50))

Upgrade(brand: VARCHAR(50) , part_number: INT, **project_id**: INT) [WEAK ENTITY]

Engineering_Team(eng_team_id: INT, team_name: VARCHAR(100), department: VARCHAR(100), HQ_address: VARCHAR(100), **eng_id**: INT)

Engineer_Assignment(eng_id: INT, name: VARCHAR(50), proficiency: VARCHAR(50), years_experience: INT, **eng_team_id**: VARCHAR(50))

University of British Columbia, Vancouver
Department of Computer Science

Functional Dependencies

Sponsor Relation

- tier_level → amount_contributed
- sponsor_id → sponsor_name, amount_contributed, tier_level, point_of_contact

Funds Relation

- **sponsor_id**, **team_id** → contract_start_date, contract_end_date

Team Relation

- team_principal → team_name
- team_id → team_name, team_principal, year_founded

Racing Series Relation

- rs_name → division, governing_body
- racing_series_id → rs_name, division, governing_body

Competes-In Relation

- **team_id**, **racing_series_id** → ranking_in_series

Vehicle Relation

- vehicle_id → model, year_first_produced, **driver_id**, **team_id**

Driver Relation

- name → driver_number
- driver_id -> name, nationality, driver_number

Project Relation

- project_id → upgrade, duration, status, **vehicle_id**, **eng_team_id**

Engineering Team Relation

- eng_team_id → team_name, department, HQ_address, **eng_id**
- team_name → HQ_address, **eng_id**

Engineer Relation

- eng_id → name, proficiency, years_experience, **eng_team_id**

University of British Columbia, Vancouver
Department of Computer Science

BCNF Normalization

Relations that were normalized

1. **Sponsor**
   Original Relation: Sponsor(sponsor_id, sponsor_name, amount_contributed, tier_level, point_of_contact)

   FDs:
- sponsor_id → sponsor_name, amount_contributed, tier_level, point_of_contact
- tier_level → amount_contributed

   Primary Key: sponsor_id
   BCNF Violation: tier_level → amount_contributed

   Decomposition:
   Sponsor_R1 (tier_level, amount_contributed)
   Primary Key: tier_level
   Sponsor_R2 (sponsor_id, sponsor_name, **tier_level**, point_of_contact)
   Primary Key: sponsor_id
   Foreign Key: tier_level (referencing Sponsor_R1)

2. **Team**
   Original Relation: Team(team_id, team_name, team_principal, year_founded)

   FDs:
- team_id → team_name, team_principal, year_founded
- team_principal → team_name

   Primary Key: team_id
   BCNF Violation: team_principal → team_name

   Decomposition:
   Team_R1 (team_principal, team_name)
   Primary Key: team_principal
   Team_R2 (team_id, **team_principal**, year_founded)
   Primary Key: team_id
   Foreign Key: team_principal (referencing Team_R1)

3. **Racing Series**
   Original Relation: Racing_series(racing_series_id, rs_name, division, governing_body)

FDs:
- racing_series_id → rs_name, division, governing_body
- rs_name → division, governing_body

Primary Key: racing_series_id
BCNF Violation: rs_name → division, governing_body

Decomposition:
Racing_Series_R1 (rs_name, division, governing_body)
Primary Key: rs_name
Racing_Series_R2 (racing_series_id, **rs_name**)
Primary Key: racing_series_id
Foreign Key: rs_name (referencing Racing_Series_R1)

4. **Driver**
Original Relation: Driver(driver_id, name, nationality, driver_number)

FDs:
- driver_id → name, nationality, driver_number
- name → driver_number

Primary Key: driver_id
BCNF Violation: name → driver_number

Decomposition:
Driver_R1 (name, driver_number)
Primary Key: name
Driver_R2 (driver_id, **name**, nationality)
Primary Key: driver_id
Foreign Key: name (referencing Driver_R1)

Relations already satisfying BCNF

Engineering_Team(eng_team_id: INT, team_name, department, HQ_address, eng_id)

Funds(**sponsor_id**: INT, **team_id**: INT, contract_start_date: DATE, contract_end_date: DATE)

Vehicle(vehicle_id: INT, model: VARCHAR(100), year_first_produced: INT, **team_id**: INT, **driver_id**: INT)

University of British Columbia, Vancouver
Department of Computer Science

Project(project_id: INT, upgrade: VARCHAR(50), duration: INT, status: VARCHAR(50), **vehicle_id**: INT, **eng_team_id**: VARCHAR(50))

Competes_In (**team_id**: INT, **racing_series_id**: INT, ranking_in_series: INT)

Engineer_Assignment(eng_id: INT, name: VARCHAR(50), proficiency: VARCHAR(50), years_experience: INT, **eng_team_id**: VARCHAR(50))

Upgrade(brand: VARCHAR(50) , part_number: INT, **project_id**: INT) [WEAK ENTITY]

University of British Columbia, Vancouver
Department of Computer Science

Final Complete Relations

So finally, renaming the decomposed relations with a meaningful name we have these final relations.

Sponsor_Tier (tier_level: VARCHAR(50), amount_contributed:INT)
- Primary Key: tier_level

Sponsor (sponsor_id: INT, sponsor_name: VARCHAR(50), **tier_level**:VARCHAR(50), point_of_contact:VARCHAR(100))
- Primary Key: sponsor_id
- Foreign Key: tier_level (referencing Sponsor_Tier)

Team_Principal (team_principal, team_name)
- Primary Key: team_principal

Team (team_id, **team_principal**, year_founded)
- Primary Key: team_id
- Foreign Key: team_principal (referencing Team_Principal)

Racing(rs_name, division, governing_body)
- Primary Key: rs_name
-

Racing_Series(racing_series_id, **rs_name**)
- Primary Key: racing_series_id
- Foreign Key: rs_name (referencing Racing)
-

Driver(name, driver_number)
- Primary Key: name
-

Driver_Internal (driver_id, **name**, nationality)
- Primary Key: driver_id
- Foreign Key: name (referencing Driver)
-

Engineering_Team(eng_team_id, team_name, department, HQ_address)
- Primary Key: eng_team_id

Engineer_Assignment(eng_id: INT, name: VARCHAR(50), proficiency: VARCHAR(50), years_experience: INT, **eng_team_id**: VARCHAR(50))
- Primary Key: eng_id
- Foreign Key: eng_team_id

Funds(**sponsor_id**: INT, **team_id**: INT, contract_start_date: DATE, contract_end_date: DATE)

- Primary Key: sponsor_id, team_id
- Foreign Key: sponsor_id, team_id

Vehicle(vehicle_id: INT, model: VARCHAR(100), year_first_produced: INT, **team_id**: INT, **driver_id**: INT)

- Primary Key: vehicle_id
- Foreign Key: team_id, driver_id

Car (**vehicle_id**: INT, drivetrain: CHAR(3), car_type: VARCHAR (50)

- Primary Key: vehicle_id

Motorcycle (**vehicle_id**: INT, engine_cc: INT, motorcycle_type: VARCHAR(50)

- Primary Key: vehicle_id

Project(project_id: INT, upgrade: VARCHAR(50), duration: INT, status: VARCHAR(50), **vehicle_id**: INT, **eng_team_id**: VARCHAR(50))

- Primary Key: project_id
- Foreign Key: vehicle_id, eng_team_id

Competes_In (**team_id**: INT, **racing_series_id**: INT, ranking_in_series: INT)

- Primary Key: team_id, racing_series_id
- Foreign Key: team_id, racing_series_id

Upgrade(brand: VARCHAR(50) , part_number: INT, **project_id**: INT) [WEAK ENTITY]

- Primary Key: brand, part_number, project_id
- Foreign Key: project_id

University of British Columbia, Vancouver
Department of Computer Science

SQL DDL Statements

Create Table Statements

```
-- Sponsor Tier Table
CREATE TABLE Sponsor_Tier (
   tier_level VARCHAR(50) PRIMARY KEY,
   amount_contributed INT
);

-- Sponsor Table
CREATE TABLE Sponsor (
   sponsor_id INT PRIMARY KEY AUTO_INCREMENT,
   sponsor_name VARCHAR(50),
   tier_level VARCHAR(50),
   point_of_contact VARCHAR(100),
   FOREIGN KEY (tier_level) REFERENCES Sponsor_Tier(tier_level)
      ON UPDATE CASCADE ON DELETE SET NULL
);
```

Foreign Key Handling Explanation

- ON UPDATE CASCADE: we need to reflect any changes made for integrity
- ON DELETE SET NULL: if a tier_level is deleted, the sponsor still exists. It wouldn't make sense to delete this.

```
-- Team Principal Table
CREATE TABLE Team_Principal (
   team_principal VARCHAR(100) PRIMARY KEY,
   team_name VARCHAR(100) UNIQUE
);

-- Team Table
CREATE TABLE Team (
   team_id INT PRIMARY KEY AUTO_INCREMENT,
   team_principal VARCHAR(100),
   year_founded INT,
   FOREIGN KEY (team_principal) REFERENCES Team_Principal(team_principal)
      ON UPDATE CASCADE ON DELETE SET NULL
);
```

Foreign Key Handling Explanation

- ON UPDATE CASCADE: we need to reflect any changes made for integrity
- ON DELETE SET NULL: even if a team has no principal, while that would be catastrophic in real life, a team would still continue to exist

```
-- Funds Table (Many-to-Many Relationship)
CREATE TABLE Funds (
   sponsor_id INT,
   team_id INT,
   contract_start_date DATE,
   contract_end_date DATE,
   PRIMARY KEY (sponsor_id, team_id),
   FOREIGN KEY (sponsor_id) REFERENCES Sponsor(sponsor_id)
      ON UPDATE CASCADE ON DELETE CASCADE,
   FOREIGN KEY (team_id) REFERENCES Team(team_id)
      ON UPDATE CASCADE ON DELETE CASCADE
);
```

Foreign Key Handling Explanation

- ON UPDATE CASCADE: we need to reflect any changes made for integrity
- ON DELETE CASCADE: in our world, a team has total participation with a sponsor (because realistically, a team cannot exist without funding from sponsors). That's why we decided that both sponsor_id and team_id would cascade if deleted.

```
-- Racing Table
CREATE TABLE Racing (
   rs_name VARCHAR(100) PRIMARY KEY,
   division VARCHAR(50),
   governing_body VARCHAR(100)
);

-- Racing Series Table
CREATE TABLE Racing_Series (
   racing_series_id INT PRIMARY KEY AUTO_INCREMENT,
   rs_name VARCHAR(100),
   FOREIGN KEY (rs_name) REFERENCES Racing(rs_name)
      ON UPDATE CASCADE ON DELETE SET NULL
);
```

Foreign Key Handling Explanation

- ON UPDATE CASCADE: we need to reflect any changes made for integrity
- ON DELETE SET NULL: if a racing series' name is deleted, it doesn't mean that the whole competition is automatically erased from history

```
-- Competes_In Table (Many-to-Many Relationship)
CREATE TABLE Competes_In (
   team_id INT,
   racing_series_id INT,
   ranking_in_series INT NOT NULL,
   PRIMARY KEY (team_id, racing_series_id),
   FOREIGN KEY (team_id) REFERENCES Team(team_id)
      ON UPDATE CASCADE ON DELETE CASCADE,
   FOREIGN KEY (racing_series_id) REFERENCES Racing_Series(racing_series_id)
      ON UPDATE CASCADE ON DELETE CASCADE
);
```

Foreign Key Handling Explanation

- ON UPDATE CASCADE: we need to reflect any changes made for integrity.
- ON DELETE CASCADE: we want to reflect active teams and racing series, not past ones. Using SET NULL would leave unwanted tuples.

```
-- Driver Table
CREATE TABLE Driver (
   name VARCHAR(100) PRIMARY KEY,
   driver_number INT NOT NULL
);

-- Driver_Internal Table
CREATE TABLE Driver_Internal (
   driver_id INT PRIMARY KEY AUTO_INCREMENT,
   name VARCHAR(100),
   nationality VARCHAR(50),
   FOREIGN KEY (name) REFERENCES Driver(name)
      ON UPDATE CASCADE ON DELETE CASCADE
);
```

Foreign Key Handling Explanation

- ON UPDATE CASCADE: we need to reflect any changes made for integrity.
- ON DELETE CASCADE: we choose to delete the driver because realistically, there can't be a driver with no name. This will leave the car with no driver open to reassignment.

```
-- Engineering Team Leads Table
CREATE TABLE Engineering_Team (
   eng_team_id INT PRIMARY KEY AUTO_INCREMENT,
   team_name VARCHAR(100),
   department VARCHAR(100),
   HQ_address VARCHAR(255),
   years_led INT
);

-- Engineer Assignment Table
CREATE TABLE Engineer_Assignment (
   eng_id INT PRIMARY KEY AUTO_INCREMENT,
   name VARCHAR(50),
   proficiency VARCHAR(50),
   years_experience INT,
   eng_team_id INT,
   FOREIGN KEY (eng_team_id) REFERENCES Engineering_Team (eng_team_id)
      ON UPDATE CASCADE ON DELETE SET NULL
);
```

Foreign Key Handling Explanation

- ON UPDATE CASCADE: we need to reflect any changes made for integrity.
- ON DELETE SET NULL: we choose to set null because engineers can always be reassigned to a different team. They don't stop existing if they don't have a team.

```
-- Vehicle Table
CREATE TABLE Vehicle (
   vehicle_id INT PRIMARY KEY AUTO_INCREMENT,
   model VARCHAR(100),
   year_first_produced INT,
   team_id INT NOT NULL,
   driver_id INT UNIQUE,
   FOREIGN KEY (team_id) REFERENCES Team(team_id)
      ON UPDATE CASCADE ON DELETE CASCADE,
   FOREIGN KEY (driver_id) REFERENCES Driver_Internal(driver_id)
      ON UPDATE CASCADE ON DELETE SET NULL
);
```

Foreign Key Handling Explanation

- ON UPDATE CASCADE: we need to reflect any changes made for integrity.
- ON DELETE CASCADE: we choose to delete the driver because realistically, there can't be a driver with no name. This will leave the car with no driver and open to reassignment.
- ON DELETE SET NULL

```
-- Car Table (ISA Subtype)
CREATE TABLE Car (
   vehicle_id INT PRIMARY KEY,
   drivetrain CHAR(3),
   car_type VARCHAR(50),
   FOREIGN KEY (vehicle_id) REFERENCES Vehicle(vehicle_id)
      ON UPDATE CASCADE ON DELETE CASCADE
);
```

Foreign Key Handling Explanation

- ON UPDATE CASCADE: we need to reflect any changes made for integrity.
- ON DELETE CASCADE: Car is an ISA subtype of Vehicles. If its vehicle_id is deleted, it's gone.

```
-- Motorcycle Table (ISA Subtype)
CREATE TABLE Motorcycle (
   vehicle_id INT PRIMARY KEY,
   engine_cc INT,
   motorcycle_type VARCHAR(50),
   FOREIGN KEY (vehicle_id) REFERENCES Vehicle(vehicle_id)
      ON UPDATE CASCADE ON DELETE CASCADE
);
```

Foreign Key Handling Explanation

- ON UPDATE CASCADE: we need to reflect any changes made for integrity.
- ON DELETE CASCADE: Motorcycle is an ISA subtype of Vehicles. If its vehicle_id is deleted, it's gone.

```
-- Project Table
CREATE TABLE Project (
   project_id INT PRIMARY KEY AUTO_INCREMENT,
   upgrade VARCHAR(50),
   duration INT,
   status VARCHAR(50),
   vehicle_id INT NOT NULL,
   eng_team_id INT,
   FOREIGN KEY (vehicle_id) REFERENCES Vehicle(vehicle_id)
      ON UPDATE CASCADE ON DELETE CASCADE,
   FOREIGN KEY (eng_team_id) REFERENCES Engineering_Team (eng_team_id)
      ON UPDATE CASCADE ON DELETE SET NULL
);
```

Foreign Key Handling Explanation

- ON UPDATE CASCADE: we need to reflect any changes made for integrity.
- ON DELETE CASCADE: we want to keep projects for CURRENT vehicles, not past vehicles. That way, engineers can see only relevant historical projects for reference if they need to make changes.
- ON DELETE SET NULL: we believe teams should be reassigned if a certain team is doing their work inefficiently or isn't doing their job properly.

```
-- Upgrade Table (Weak Entity)
CREATE TABLE Upgrade (
   brand VARCHAR(50),
   part_number INT,
   project_id INT,
   PRIMARY KEY (brand, part_number, project_id),
   FOREIGN KEY (project_id) REFERENCES Project(project_id)
      ON UPDATE CASCADE ON DELETE CASCADE
);
```

Foreign Key Handling Explanation

- ON UPDATE CASCADE: we need to reflect any changes made for integrity.
- ON DELETE CASCADE: this is a weak entity. If the project is deleted, then an upgrade cannot exist.

University of British Columbia, Vancouver
Department of Computer Science

Insert Statements

```
-- Sponsor_Tier
INSERT INTO Sponsor_Tier (tier_level, amount_contributed) VALUES
('Bronze', 100000),
('Silver', 500000),
('Gold', 1000000),
('Platinum', 5000000),
('Diamond', 10000000);

-- Sponsor
INSERT INTO Sponsor (sponsor_name, tier_level, point_of_contact) VALUES
('Red Bull', 'Gold', 'John Browning'),
('Petronas', 'Platinum', 'Sarah Johnson'),
('Pirelli', 'Silver', 'Michael Smith'),
('Shell', 'Diamond', 'Anna Brown'),
('Honda', 'Bronze', 'David Wilson');

-- Team_Principal
INSERT INTO Team_Principal (team_principal, team_name) VALUES
('Toto Wolff', 'Mercedes AMG Petronas F1 Team'),
('Christian Horner', 'Red Bull Racing'),
('Zak Brown', 'McLaren Racing'),
('Laurent Rossi', 'Alpine Racing Co.'),
('Mattia Binotto', 'Scuderia Corsa'),
('Nazif Ishrak', 'The KTM Factory'),
('Eddie Kim', 'Ducati Racing Team');

-- Team
INSERT INTO Team (team_principal, year_founded) VALUES
('Toto Wolff', 1954),
('Christian Horner', 2005),
('Zak Brown', 1963),
('Laurent Rossi', 1975),
('Mattia Binotto', 1929),
('Nazif Ishrak', 2025),
('Eddie Kim', 2023);

-- Racing
INSERT INTO Racing (rs_name, division, governing_body) VALUES
('Formula 1', 'Open Wheel', 'FIA'),
('MotoGP', 'Motorcycle', 'FIM'),
('WEC', 'Endurance', 'ACO'),
```

```
('IndyCar', 'Open Wheel', 'INDYCAR'),
('WRC', 'Rally', 'FIA');

-- Racing_Series
INSERT INTO Racing_Series (rs_name) VALUES
('Formula 1'),
('MotoGP'),
('WEC'),
('IndyCar'),
('WRC');

-- Driver
INSERT INTO Driver (name, driver_number) VALUES
('Lewis Hamilton', 44),
('Max Verstappen', 33),
('Charles Leclerc', 16),
('Fernando Alonso', 14),
('Sebastian Vettel', 5),
('Marc Marquez', 93),
('Jorge Martin', 1),
('Fabio Quartararo', 20),
('Joan Mir', 36),
('Daniel Owen Santosa', 9);


-- Driver_Internal
INSERT INTO Driver_Internal (name, nationality) VALUES
('Lewis Hamilton', 'British'),
('Max Verstappen', 'Dutch'),
('Charles Leclerc', 'Monégasque'),
('Fernando Alonso', 'Spanish'),
('Sebastian Vettel', 'German'),
('Marc Marquez', 'Spanish'),
('Jorge Martin', 'Spanish'),
('Fabio Quartararo', 'French'),
('Joan Mir', 'Spanish'),
('Daniel Owen Santosa', 'Indonesian');

-- Engineering_Team
INSERT INTO Engineering_Team (team_name, department, HQ_address, years_led) VALUES
('Mercedes', 'Aerodynamics', 'Brackley, UK', 5),
('Red Bull', 'Powertrain', 'Milton Keynes, UK', 7),
('McLaren', 'Chassis', 'Woking, UK', 3),
```

```
('Ferrari', 'Electronics', 'Maranello, Italy', 6),
('Alpine', 'Strategy', 'Enstone, UK', 4);

-- Engineer_Assignment
INSERT INTO Engineer_Assignment (name, proficiency, years_experience) VALUES
('John Engineer', 'Aerodynamics', 10),
('Alice Tech', 'Powertrain', 8),
('Bob Mechanic', 'Chassis', 12),
('Clara Data', 'Electronics', 9),
('Daniel Strategist', 'Strategy', 11),
('John McLane', 'Aerodynamics', 2);

-- Funds
INSERT INTO Funds (sponsor_id, team_id, contract_start_date, contract_end_date) VALUES
(1, 1, '2023-01-01', '2025-12-31'),
(2, 2, '2022-06-13', '2024-05-31'),
(3, 3, '2021-03-20', '2023-12-31'),
(4, 4, '2020-07-30', '2025-07-01'),
(5, 5, '2022-11-02', '2024-10-31'),
(5, 2, '2023-05-10', '2024-12-31');

-- Vehicle
INSERT INTO Vehicle (model, year_first_produced, team_id, driver_id) VALUES
('Mercedes W14', 2023, 1, 1),
('Red Bull RB19', 2023, 2, 2),
('Ferrari 488 GT3', 2018, 5, 3),
('McLaren F1 GTR', 1995, 3, 4),
('Alpine A523', 2023, 1, 5),
('Suzuki GSX-RR', 2010, 1, 6),
('KTM RC16', 2015, 3, 7),
('Honda RC213V-RS', 2023, 2, 8),
('Ducati Desmosedici', 2010, 5, 9),
('Ducati Diavel', 2003, 4, 10);

-- Car
INSERT INTO Car (vehicle_id, drivetrain, car_type) VALUES
(1, 'AWD', 'Formula 1'),
(2, 'AWD', 'Formula 1'),
(3, 'RWD', 'Grand Tourer'),
(4, 'AWD', 'Grand Tourer'),
(5, '4WD', 'Formula 1');

-- Motorcycle
```

```
INSERT INTO Motorcycle (vehicle_id, engine_cc, motorcycle_type) VALUES
(6, 1000, 'MotoGP'),
(7, 850, 'MotoGP'),
(8, 500, 'Cruiser'),
(9, 850, 'Touring'),
(10, 450, 'Dirtbike');

-- Project
INSERT INTO Project (upgrade, duration, status, vehicle_id, eng_team_id) VALUES
('New Aero Package', 6, 'Ongoing', 1, 1),
('Power Unit Upgrade', 4, 'Planned', 2, 2),
('Lightweight Chassis', 5, 'Completed', 3, 3),
('Hybrid System Improvement', 7, 'Ongoing', 4, 3),
('Brake System Redevelopment', 3, 'Completed', 5, 4),
('New Brake Calipers', 2, 'Completed', 6, 5),
('Tail-End Aero Package Retrofitment', 1, 'Completed', 6, 3),
('Exhaust Redesign', 5, 'Planned', 7, 5);

-- Competes_In
INSERT INTO Competes_In (team_id, racing_series_id, ranking_in_series) VALUES
(1, 1, 1),
(2, 1, 2),
(3, 1, 3),
(4, 1, 4),
(4, 2, 1),
(5, 1, 5),
(5, 3, 2),
(6, 2, 3),
(7, 2, 4);

-- Upgrade
INSERT INTO Upgrade (brand, part_number, project_id) VALUES
('Brembo', 101, 1),
('Bosch', 102, 2),
('Michelin', 103, 3),
('Pirelli', 104, 4),
('Honda', 105, 5),
('Honda', 900, 2),
('Royal Purple', 001, 5),
('EBC', 1222, 6),
('Seibon Carbon', 1000, 7),
('Magnaflow', 20032, 8);
```