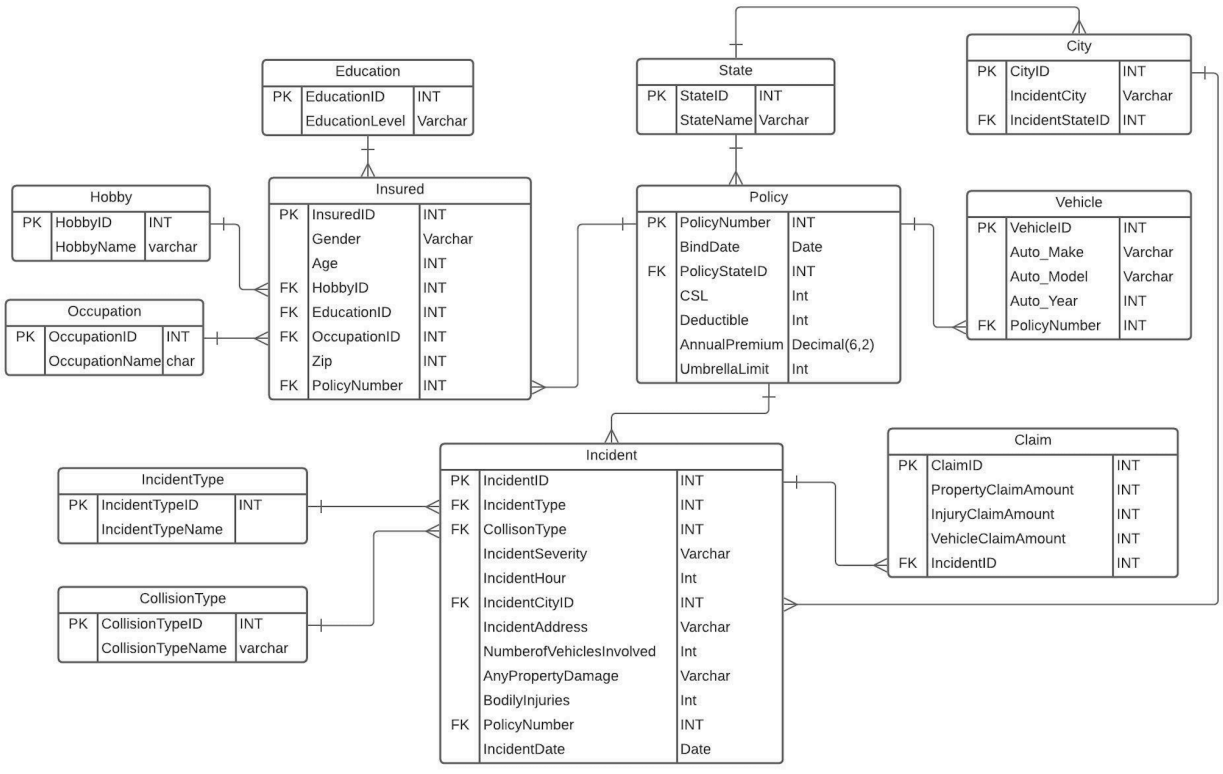
Database Project for a Car Insurance Company

This project consists of 3 parts, including:
part I: data modeling
Part II: create a relational database, inputting the data from existing tables and forms
Part III: write queries to answer business questions

Part I: data modeling

ERD



Part II: create a relational database

```
create table cl_customer(
	CustomerID INT AUTO_INCREMENT PRIMARY KEY,
	Gender varchar(50),
	Age varchar(25),
	EducationID INT,
	HobbyID INT,
	OccupationID INT,
	ZipID INT,
	FOREIGN KEY (EducationID) REFERENCES cl_education (EducationID),
	FOREIGN KEY (HobbyID) REFERENCES cl_hobby (HobbyID),
	FOREIGN KEY (OccupationID) REFERENCES cl_occupation (OccupationID),
	FOREIGN KEY (ZipID) REFERENCES cl_zip (ZipID)
)

INSERT INTO occupation(OccupationName)
SELECT DISTINCT`insured_occupation` FROM FP_Claims.`Claims`

insert INTO cl_city(CityName, StateID)
SELECT
	DISTINCT `incident_city`,
	StateID
FROM
	cl_state s,
	FP_Claims.`Claims` c
WHERE
	s.StateName = c.StateName

SELECT
	DISTINCT insured_zip,
	CityID
FROM
	cl_city c,
	FP_Claims.`Claims` cl
WHERE
	c.CityName = cl.incident_city


SELECT
	DISTINCT `insured_sex`,
	`age`, policy_number,
```

```
        HobbyID,
        EducationID,
        OccupationID,
    insured_zip
FROM
        cl_hobby h,
    cl_education e,
    cl_occupation o,
    FP_Claims.Claims c
WHERE
h.HobbyName = c.insured_hobbies
AND e.EducationLevel = c.`insured_education_level`
AND o.OccupationName = c.`insured_occupation`



INSERT INTO cl_vehicle (`Auto_Model`,`Auto_Make`,`Auto_Year`,`PolicyNumber`)
SELECT
        DISTINCT `auto_model`,
        `auto_make`,
    `auto_year`,
    PolicyNumber

FROM
    cl_policy p,
        FP_Claims.`Claims` c
WHERE
    p.PolicyNumber = c.policy_number



    CREATE VIEW FinalProject_claims_incidentstate_view AS
    SELECT
        ClaimID,
        p.PolicyNumber,
        BindDate,
        CSL,
        Deductible,
        AnnualPremium,
        UmbrellaLimit,
        i.IncidentID,
        IncidentDate,
        IncidentTypeName,
```

```
        CollisionTypeName,
        IncidentSeverity,
        IncidentHour,
        IncidentAddress,
        cl_city.IncidentCity,
        StateName as IncidentState,
        Number_of_Vehicles_Involved,
        PropertyDamage,
        BodilyInjuries,
        VehicleID,
        Auto_Model,
        Auto_Make,
        Auto_Year,
        InsuredID,
        Gender,
        Age,
        Zip,
        HobbyName,
        EducationLevel,
        OccupationName,
        PropertyClaimAmount,
        InjuryClaimAmount,
        VehicleClaimAmount
FROM
        cl_CollisionType ct,
        cl_IncidentType it,
        cl_city,
        cl_claim cl,
        cl_education e,
        cl_hobby h,
        cl_incident i,
        cl_insured ins,
        cl_occupation o,
        cl_policy p,
        cl_state s,
        cl_vehicle v
WHERE
        ins.EducationID = e.EducationID
        AND ins.HobbyID = h.HobbyID
        AND ins.OccupationID = o.OccupationID
        AND ins.PolicyNumber = p.PolicyNumber
        AND i.IncidentType = it.IncidentTypeID
        AND i.CollisionType = ct.CollisionTypeID
        AND i.IncidentCity = cl_city.CityID
```

```
    AND i.PolicyNumber = p.PolicyNumber
    AND cl_city.IncidentStateID = s.StateID
    AND cl.IncidentID = i.IncidentID
    AND v.PolicyNumber = p.PolicyNumber
ORDER BY 1
```

_____

```
CREATE VIEW FinalProject_claims_policystate_view AS
SELECT
    p.PolicyNumber,
    StateName as PolicyState
FROM
    cl_policy p,
    cl_state s
WHERE
    p.PolicyStateID = s.StateID
```

_____

Join two views to form a new view which is used for querry:

```
CREATE VIEW FinalProject_claims_view AS
SELECT
    `ClaimID`,
    v1.`PolicyNumber`,
    `BindDate`,
    `PolicyState`,
    `CSL`,
    `Deductible`,
    `AnnualPremium`,
    `UmbrellaLimit`,
    `IncidentID`,
    `IncidentDate`,
    `IncidentTypeName`,
    `CollisionTypeName`,
    `IncidentSeverity`,
    `IncidentHour`,
    `IncidentAddress`,
    `IncidentCity`,
    `IncidentState`,
    `Number_of_Vehicles_Involved`,
    `PropertyDamage`,
    `BodilyInjuries`,
    `VehicleID`,
    `Auto_Model`,
```

```
    `Auto_Make`,
    `Auto_Year`,
    `InsuredID`,
    `Gender`,
    `Age`,
    `Zip`,
    `HobbyName`,
    `EducationLevel`,
    `OccupationName`,
    `PropertyClaimAmount`,
    `InjuryClaimAmount`,
    `VehicleClaimAmount`
FROM
    `FinalProject_claims_incidentstate_view` v1
    JOIN FinalProject_claims_policystate_view v2 ON
  v1.PolicyNumber = v2.PolicyNumber
```

Part III: Queries

1. Display total claim amount by year and by month

```
SELECT
        Year(`IncidentDate`),
        Date_Format(`IncidentDate`,'%M') AS Month,
        SUM(PropertyClaimAmount)+ SUM(`InjuryClaimAmount`)+ SUM(`InjuryClaimAmount`)
as TotalClaimAmount
FROM
        `FinalProject_claims_view`
GROUP BY 1, 2
ORDER BY 3 DESC
```

2. Display total claim amount by PolicyState

```
SELECT
        PolicyState,
        SUM(PropertyClaimAmount)+ SUM(InjuryClaimAmount)+ SUM(InjuryClaimAmount) as
TotalClaimAmount
FROM
        `FinalProject_claims_view`
GROUP BY 1
```

3. Display number of policies by year and by state
(How many new policies are generated by each state in each year?)

```
SELECT
   year(BindDate),
   PolicyState,
   COUNT(*)
FROM
        `FinalProject_claims_view`
GROUP BY 1,2
```

4. Display number of incidents by gender (Does gender affect number of incidents?)

```
SELECT
        Gender,
        Count(*)
FROM
```

```
    FinalProject_claims_view
GROUP BY 1
```

5. Display number of incidents by age (Which age is prone to have car incidents?)

```
SELECT
    Age,
    Count(*)
FROM
    FinalProject_claims_view
GROUP BY 1
ORDER BY 2 DESC
```

6. Display number of incidents by Auto_Year (Does the age of vehicle affect the number of incidents?)

```
SELECT
    Auto_Year,
    Year(IncidentDate)- Auto_Year as Vehicle_Age,
    COUNT(*)
FROM
    FinalProject_claims_view
GROUP BY 1, 2
ORDER BY 3 DESC
```

7. Display number of incidents by Auto_Model (Which vehicle Model tend to have more incidents?)

```
SELECT
    Auto_Make,
    Auto_Model,
    COUNT(*)
FROM
    FinalProject_claims_view
GROUP BY 1, 2
ORDER BY 3 DESC
```

8. Display number of claims by occupation (Does occupation affect claims?)

```
SELECT
    OccupationName,
    COUNT(*)
FROM
    FinalProject_claims_view
```

```
GROUP BY 1
ORDER BY 2 DESC
```

9. Display number of claims by education (Does education affect claims?)

```
SELECT
    EducationLevel,
    COUNT(*)
FROM
    FinalProject_claims_view
GROUP BY 1
ORDER BY 2 DESC
```

10. Display number of claims by incident hour (In which hour of the day does most incidents occur?)

```
SELECT
    IncidentHour,
    COUNT(*)
FROM
    FinalProject_claims_view
GROUP BY 1
ORDER BY 1
```

11. Display number of ‘total loss’ claims by age (Drivers at which age are more likely to have a ‘total loss’ incident?)

```
SELECT
    IncidentSeverity,
    Age as Driver_Age,
    COUNT(*)
FROM
    FinalProject_claims_view
WHERE
    IncidentSeverity = 'Total Loss'
GROUP BY 1
ORDER BY 2 DESC
```

12. Display number of claims by collision type (Which collision type is more frequent?)

```
SELECT
CollisionTypeName,
    COUNT(*)
FROM
```

```
        FinalProject_claims_view
    GROUP BY 1
    ORDER BY 2 DESC
```

13. Display number of incidents by incident type (Which incidents type is more frequent?)

```
    SELECT
        IncidentTypeName,
        COUNT(*)
    FROM
        FinalProject_claims_view
    GROUP BY 1
    ORDER BY 2 DESC
```

14. Display number of incidents by Incident State

```
    SELECT
        IncidentStateName,
        COUNT(*)
    FROM
        FinalProject_claims_view
    GROUP BY 1
    ORDER BY 2 DESC
```

15. Display Total Revenue by policy state

```
        SELECT
                PolicyState,
               SUM(AnnualPremium) TotalRevenue
        FROM
                `FinalProject_claims_view`
        GROUP BY 1
        ORDER BY 2
```

16. Display gain/loss by PolicyState and by Year (Which policy state has most gain?)

```
    SELECT
        year(IncidentDate),
        PolicyState,
      SUM(AnnualPremium) TotalRevenue,
      SUM(PropertyClaimAmount)+ SUM(InjuryClaimAmount)+ SUM(VehicleClaimAmount) as
TotalClaimAmount,
      SUM(AnnualPremium)- (SUM(PropertyClaimAmount)+ SUM(InjuryClaimAmount)+
SUM(VehicleClaimAmount)) as Gain_or_Loss
    FROM
        `FinalProject_claims_view`
```

GROUP BY 1, 2
order BY 2