**Schema :**

1. **‘workers’** table stores information about workers

CREATE TABLE workers (

worker\_id INT PRIMARY KEY AUTO\_INCREMENT,

first\_name VARCHAR(50) NOT NULL,

last\_name VARCHAR(50) NOT NULL,

date\_of\_birth DATE,

occupation VARCHAR(50),

industry\_type VARCHAR(50)

);

1. **‘claims’** table stores data related to workers' compensation claims, including a foreign key(worker\_id) relationship to the **‘workers’** table

CREATE TABLE claims (

claim\_id INT PRIMARY KEY AUTO\_INCREMENT,

worker\_id INT,

claim\_date DATE,

closed\_date DATE,

report\_date DATE,

claim\_cost DECIMAL(15, 2),

loss\_type VARCHAR(50),

litigation\_status VARCHAR(50),

cause\_description TEXT,

FOREIGN KEY (worker\_id) REFERENCES workers(worker\_id)

);

1. **‘model\_predictions’** table stores predictions from machine learning models, including a foreign key relationship to the **‘claims’** table

CREATE TABLE model\_predictions (

prediction\_id INT PRIMARY KEY AUTO\_INCREMENT,

claim\_id INT,

predicted\_claim\_cost DECIMAL(15, 2),

prediction\_date DATE,

model\_version VARCHAR(20),

FOREIGN KEY (claim\_id) REFERENCES claims(claim\_id)

);

**SQL Queries**:

1. Retrieve workers' details along with their total claimed costs –

SELECT

w.worker\_id,

w.first\_name,

w.last\_name,

w.date\_of\_birth,

w.occupation,

w.industry\_type,

SUM(c.claim\_cost) AS total\_claimed\_cost

FROM

workers w

LEFT JOIN

claims c ON w.worker\_id = c.worker\_id

GROUP BY

w.worker\_id, w.first\_name, w.last\_name, w.date\_of\_birth, w.occupation, w.industry\_type;

1. Calculate average claim costs based on industry types

SELECT

w.industry\_type,

AVG(c.claim\_cost) AS average\_claim\_cost

FROM

workers w

JOIN

claims c ON w.worker\_id = c.worker\_id

GROUP BY

w.industry\_type;