Project 1: Part B Answers

B.1. Schema Normalization

The following is a relation schema from a real-world sales application at a major retail company:

**Sales** (Store, Dept, WeekDate, WeeklySales, IsHoliday, Type, Size, Temperature, FuelPrice, CPI, UnemploymentRate)

The following Functional Dependencies (FDs) are given to hold on **Sales**:

Store, WeekDate → Store, Temperature, Size, FuelPrice, CPI, UnemploymentRate, WeekDate, Type
Size, Store, WeekDate → Type, IsHoliday
Store → Type, Size
Store, Dept, WeekDate, IsHoliday → Temperature, CPI, Type, IsHoliday
Store, Dept, WeekDate → WeeklySales, WeekDate, Dept
WeekDate → IsHoliday

1. **[10%]** Compute the canonical (minimal) cover of the set of FDs on **Sales** given above.
   
   **Answer:** The canonical cover here is unique and has exactly the following 4 FDs.
   
   WeekDate → IsHoliday
   Store → Type, Size
   Store, WeekDate → Temperature, FuelPrice, CPI, UnemploymentRate
   Store, Dept, WeekDate → WeeklySales

2. **[10%]** Perform BCNF decomposition of **Sales** using the canonical cover. Name your decomposed relations appropriately, along with the keys and foreign keys in each relation. Preserve the attribute names.

   **Answer:** The BCNF decomposition is unique here and is as follows.

   Holidays (WeekDate, IsHoliday)
   Stores (Store, Type, Size)
   TemporalData (Store, WeekDate, Temperature, FuelPrice, CPI, UnemploymentRate)
   SalesNew (Store, Dept, WeekDate, WeeklySales)

   Each relation has a unique key, and it is underlined in its schema.
   Two relations have foreign keys: TemporalData and SalesNew.
   TemporalData has two foreign keys: WeekDate is a foreign key referring to Holidays, and Store is a foreign key referring to Stores.
SalesNew has three foreign keys: WeekDate is a foreign key referring to Holidays, Store is a foreign key referring to Stores, and {Store, WeekDate} is a foreign key referring to TemporalData.

3. [10%] Write SQL commands (for SQLite3 DBMS) to create the relations in the BCNF schema. The attribute data types are listed in the table below. Specify all the appropriate key and foreign key constraints.

Answer: The SQL queries are as follows.

```sql
CREATE TABLE Holidays (  
    WeekDate DATE PRIMARY KEY,  
    IsHoliday BOOLEAN
);  
CREATE TABLE Stores (  
    Store INTEGER PRIMARY KEY,  
    Type CHAR(1),  
    Size INTEGER
);  
CREATE TABLE TemporalData (  
    Store INTEGER,  
    WeekDate DATE,  
    Temperature REAL,  
    FuelPrice REAL,  
    CPI REAL,  
    UnemploymentRate REAL,  
    PRIMARY KEY (Store, WeekDate),  
    FOREIGN KEY(Store) REFERENCES Stores(Store),  
    FOREIGN KEY(WeekDate) REFERENCES Holidays(WeekDate)
);  
CREATE TABLE SalesNew (  
    Store INTEGER,  
    Dept INTEGER,  
    WeekDate DATE,  
    WeeklySales REAL,  
    PRIMARY KEY (Store, Dept, WeekDate),  
    FOREIGN KEY(Store) REFERENCES Stores(Store),  
    FOREIGN KEY(WeekDate) REFERENCES Holidays(WeekDate),  
    FOREIGN KEY(Store, WeekDate)  
        REFERENCES TemporalData(Store, WeekDate)
);  
```
<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Store</td>
<td>Integer</td>
</tr>
<tr>
<td>Dept</td>
<td>Integer</td>
</tr>
<tr>
<td>WeekDate</td>
<td>Date</td>
</tr>
<tr>
<td>WeeklySales</td>
<td>Real</td>
</tr>
<tr>
<td>IsHoliday</td>
<td>Boolean</td>
</tr>
<tr>
<td>Type</td>
<td>Char(1)</td>
</tr>
<tr>
<td>Size</td>
<td>Integer</td>
</tr>
<tr>
<td>Temperature</td>
<td>Real</td>
</tr>
<tr>
<td>FuelPrice</td>
<td>Real</td>
</tr>
<tr>
<td>CPI</td>
<td>Real</td>
</tr>
<tr>
<td>UnemploymentRate</td>
<td>Real</td>
</tr>
</tbody>
</table>

### B.2. Instance Normalization

[20%] Write code in either C++, Java, R, Perl, or Python (the latter three might be easier) to convert the instance in `SalesFile.csv` into the individual relation instances of the relations in the BCNF schema that you obtained from portion B.1.2. Print out the decomposed relation instances in the CSV plaintext file format just like `SalesFile.csv` (the first line must be a list of the attribute names). Use the same names as your relation names for your CSV files.

**Answer:** The 4 relations in the normalized output should have the same schema as in the answer to B.1.2. The CSV files corresponding to the normalized instances must have the following numbers of tuples. Thus, the code must deduplicate the tuples.

<table>
<thead>
<tr>
<th>Relation</th>
<th>Number of Tuples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Holidays</td>
<td>143</td>
</tr>
<tr>
<td>Stores</td>
<td>45</td>
</tr>
<tr>
<td>TemporalData</td>
<td>6435</td>
</tr>
<tr>
<td>SalesNew</td>
<td>421570</td>
</tr>
</tbody>
</table>