**Housing Inspection Explorer: A Tool for Viewing Inspection Scores Among Public and Multihousing Units**

**Team Name:** Housing Inspection Explorer Team

**Project Type:** Simple Project

**Team Members:**

* Phillips, Christopher Trevor [ctp180001]
* Nazneen, Amtul [axn180041]

**1. INTRODUCTION**

The Housing Inspection Explorer (HIE) provides insight into the Multifamily and Public Housing Inspection Scores published by the Public Housing Authority (PHA). PHA performs the inspections for metropolitan and micropolitan statistical areas, which are geographical entities defined by the U.S. Office of Management and Budget (OMB).

HIE has features which allows users to view all inspections scores within a particular state, locate the top states and counties with highest or lowest aggregated inspection scores, and all inspections that were performed in a given year. It also gives insight into cities that have acceptable Housing Inspection Scores, and the estimated next inspection dates for specific housing properties. Thus, the tool provides an intuitive means to view and analyze Housing Inspection Scores in a convenient graphical reporting interface.

**2. TARGET AUDIENCE**

HIE gives access to the published PHA Housing Inspection Scores through interactive dynamic reports and can be used by agencies to publish Federal statistics. HIE also assists potential home buyers and property owners to determine if a given location is decent, safe, sanitary and in good repair. Thus, the target audience for the HIE is comprised of both government agents/representatives and independent citizens, both of whom gauge the “wellness” of housing properties for their various respective purposes.

**3. DESCRIPTION OF DATA SOURCES**

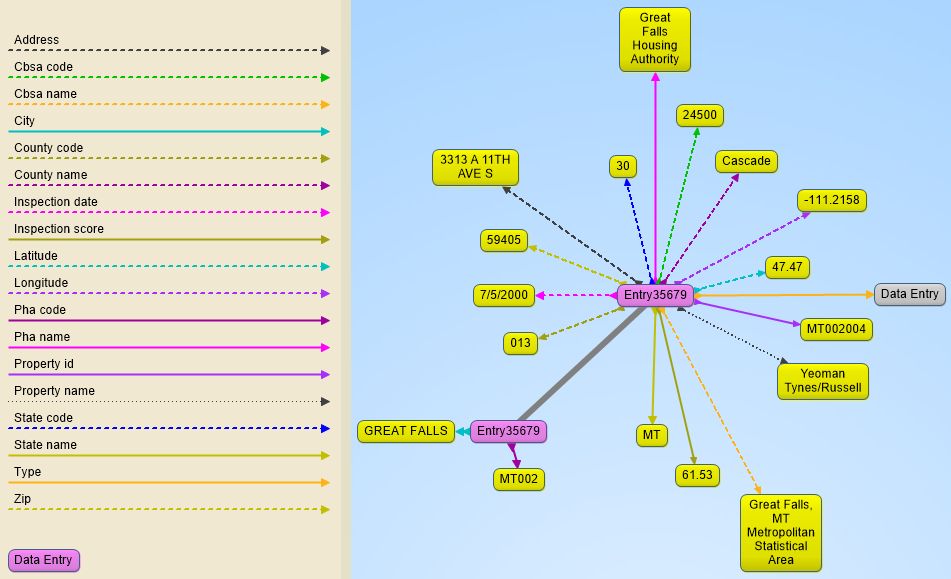
Datasets under consideration are:

* Dataset 1258- [Public Housing Physical Inspection Scores](http://data-gov.tw.rpi.edu/vocab/Dataset_1258) (No. of triples: 1278872)
* Dataset 1259- [Multifamily Housing Physical Inspection Scores](http://data-gov.tw.rpi.edu/vocab/Dataset_1259) (No. of triples: 1455325)

Housing properties are identified using a unique property identification number (or Property ID), and each housing property is associated with various attributes such as location (address, city etc.), inspection date, and inspection score. Thus, each triple in both datasets has the following ontological form:

**[Property ID] -> [Attribute] -> [Value]**

An example of this can be seen for a single housing property below. The housing property identified by “Entry35679” acts as the subject for several triples, each of which has a relevant attribute and associated value (e.g. [Entity35879] -> [Zip] -> [59405]).



*Figure 1: A single property with all associated attributes and values.*

Full descriptions of all attributes in both datasets are given in the tables below.

**Dataset 1258 – Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| Property\_id | Property identification number |
| Property\_name | Full property name |
| address | Property street address |
| city | Property city name |
| Cbsa\_name | Core-based statistical area name |
| Cbsa\_code | Core-based statistical area code |
| County\_name | County name |
| County\_code | County FIPS code |
| State\_name | State postal abbreviation |
| State\_code | State FIPS code |
| zip | ZIP Code |
| latitude | Property latitude |
| longtitude | Property longitude |
| Pha\_code | Public Housing Authority code |
| Pha\_name | Formal PHA name |
| Inspection\_score | Overall inspection score |
| Inspection\_date | Date of inspection |

**Dataset 1259 – Attributes**

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| Property\_id | Property identification number |
| Property\_name | Full property name |
| address | Property street address |
| city | Property city name |
| Cbsa\_name | Core-based statistical area name |
| Cbsa\_code | Core-based statistical area code |
| County\_name | County name |
| County\_code | County FIPS code |
| State\_name | State postal abbreviation |
| State\_code | State FIPS code |
| zip | ZIP Code |
| latitude | Property latitude |
| longtitude | Property longitude |
| Inspection\_score | Overall inspection score |
| Inspection\_date | Date of inspection |

**4. DATA INTEGRATION**

The next step was to integrate both datasets in order to leverage their combined information. However, the SPARQL endpoint on the LOGD website (where the datasets originally resided) stopped functioning, so it was necessary to create our own SPARQL endpoint using Apache Jena Fuseki. Thus, both datasets were downloaded in their raw N-Triples formats and uploaded to a local instance of Apache Jena Fuseki server. This provided a local SPARQL endpoint by which the HIE could query the datasets.

**Combining Datasets**

Dataset 1258 and 1259: Upon integration, we were able to derive overall information about the inspection scores across the U.S. for both the Public Housing and Multifamily Housing Datasets by combining on a shared attribute. We accomplished this by combining on the “State\_name” attribute common to both datasets, which lists the state associated with a housing property (e.g. “CA”, “TX”, etc.).

|  |  |
| --- | --- |
| **Join Attribute** | **Description** |
| State\_name | State Postal Abbreviation |

**5. DATA PRODUCT RESULTS**

**5.1 SPARQL Queries**

By integrating with the ‘**state\_name**’ attribute, SPARQL Queries were executed against the Fuseki SPARQL Endpoint. SPARQL Queries along with their expected results are discussed below.

**Query: Average Scores by State**

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT ?state (AVG(xsd:float(?inspection\_score)) AS ?score)

FROM <http://data-gov.tw.rpi.edu/vocab/Dataset\_1258>

FROM <http://data-gov.tw.rpi.edu/vocab/Dataset\_1259>

WHERE

{

{

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1258/state\_name> ?state.

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1258/inspection\_score> ?inspection\_score.

}

UNION

{

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1259/state\_name> ?state.

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1259/inspection\_score> ?inspection\_score.

}

}

GROUP BY ?state

ORDER BY ?state

**Result**

Gives average Housing Inspection Scores for each state (that is, the scores obtained by averaging all Housing Inspection Scores within the state).

**Query: Average Scores by County**

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT ?state ?county (AVG(xsd:float(?inspection\_score)) AS ?score)

FROM <http://data-gov.tw.rpi.edu/vocab/Dataset\_1258>

FROM <http://data-gov.tw.rpi.edu/vocab/Dataset\_1259>

WHERE

{

{

SELECT ?state (LCASE(?county\_name\_mc) as ?county) (LCASE(?city\_mc) as ?city) ?inspection\_score

WHERE{

{

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/state\_name> ?state.

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/county\_name> ?county\_name\_mc.

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/inspection\_score> ?inspection\_score.

}

UNION

{

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/state\_name> ?state.

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/county\_name> ?county\_name\_mc.

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/inspection\_score> ?inspection\_score.

}

}

}

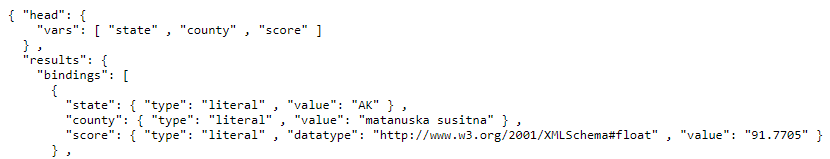
}

GROUP BY ?state ?county

ORDER BY ?state ?county

**Result**

Gives average Housing Inspection Scores for all counties, grouped by state (that is, the scores obtained by averaging all Housing Inspection Scores within the county, grouped by state). A similar query is used for calculate average scores for all cities grouped by state by replacing all instances of county with city within the query.



**Query: Inspection Information**

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT ?city ?county\_name ?inspection\_score ?inspection\_date

FROM <http://data-gov.tw.rpi.edu/vocab/Dataset\_1259>

FROM <http://data-gov.tw.rpi.edu/vocab/Dataset\_1258>

WHERE{

{

SELECT ?state\_name ?inspection\_score ?inspection\_date (CONCAT(UCASE(SUBSTR(?city\_mc, 1, 1)), LCASE(SUBSTR(?city\_mc, 2))) as ?city) (CONCAT(UCASE(SUBSTR(?county\_name\_mc, 1, 1)), LCASE(SUBSTR(?county\_name\_mc, 2))) as ?county\_name)

WHERE{

{

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/state\_name> ?state\_name FILTER regex(?state\_name, "state") .

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/city> ?city\_mc.

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/county\_name> ?county\_name\_mc.

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/inspection\_score> ?inspection\_score.

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/inspection\_date> ?inspection\_date

FILTER regex(?inspection\_date, "year")

} UNION

{

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/state\_name>?state\_name FILTER regex(?state\_name, "state") .

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/city> ?city\_mc.

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/county\_name> ?county\_name\_mc.

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/inspection\_score> ?inspection\_score.

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/inspection\_date> ?inspection\_date

FILTER regex(?inspection\_date, "year")

}

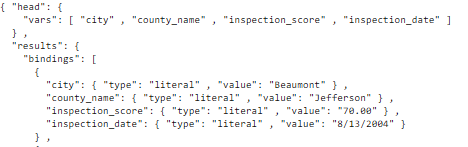
}

}

}

**Result**

Gives information about all the inspections that happened in a given year and for a given state.



**Query: Average Scores, Latest Inspection Dates by Property**

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT ?state ?county ?city ?property (AVG(xsd:float(?inspection\_score)) AS ?score) (MAX(?inspection\_date) AS ?date)

FROM <http://data-gov.tw.rpi.edu/vocab/Dataset\_1258>

FROM <http://data-gov.tw.rpi.edu/vocab/Dataset\_1259>

WHERE

{

{

SELECT ?state (LCASE(?county\_name\_mc) as ?county) (LCASE(?city\_mc) as ?city) (LCASE(?property\_name\_mc) as ?property) ?inspection\_date ?inspection\_score

WHERE{

{

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/state\_name> ?state.

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/county\_name> ?county\_name\_mc.

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/city> ?city\_mc.

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/property\_name> ?property\_name\_mc.

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/inspection\_date> ?inspection\_date.

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/inspection\_score> ?inspection\_score.

}

UNION

{

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/state\_name> ?state.

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/county\_name> ?county\_name\_mc.

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/city> ?city\_mc.

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/property\_name> ?property\_name\_mc.

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/inspection\_date> ?inspection\_date.

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/inspection\_score> ?inspection\_score.

}

}

}

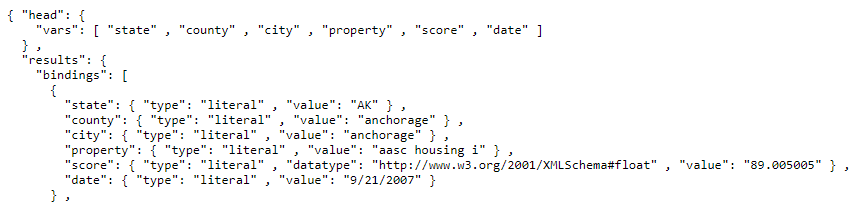
}

GROUP BY ?state ?county ?city ?property

ORDER BY ?state ?county ?city ?property

**Result**

Gives average Housing Inspection Scores and latest inspection dates for all properties, grouped by state, county, and city.



**Query: Top/Bottom 5 Scoring States**

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT ?state\_name (COUNT(\*) AS ?howmany) (AVG(xsd:float(?inspection\_score)) AS ?score)

FROM <http://data-gov.tw.rpi.edu/vocab/Dataset\_1259>

FROM <http://data-gov.tw.rpi.edu/vocab/Dataset\_1258>

WHERE{

{

SELECT ?state\_name ?inspection\_score (LCASE(?city\_mc) as ?city) (LCASE(?county\_name\_mc) as ?county\_name)

WHERE{

{

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/state\_name> ?state\_name.

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/inspection\_score> ?inspection\_score.

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/city> ?city\_mc.

?s1 <http://data-gov.tw.rpi.edu/vocab/p/1259/county\_name> ?county\_name\_mc.

}

UNION

{

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/state\_name> ?state\_name.

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/inspection\_score> ?inspection\_score.

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/city> ?city\_mc.

?s2 <http://data-gov.tw.rpi.edu/vocab/p/1258/county\_name> ?county\_name\_mc.

}

}

}

}

GROUP BY ?state\_name

HAVING (?howmany >= 100)

ORDER BY DESC(?score)

LIMIT 5

**Result**

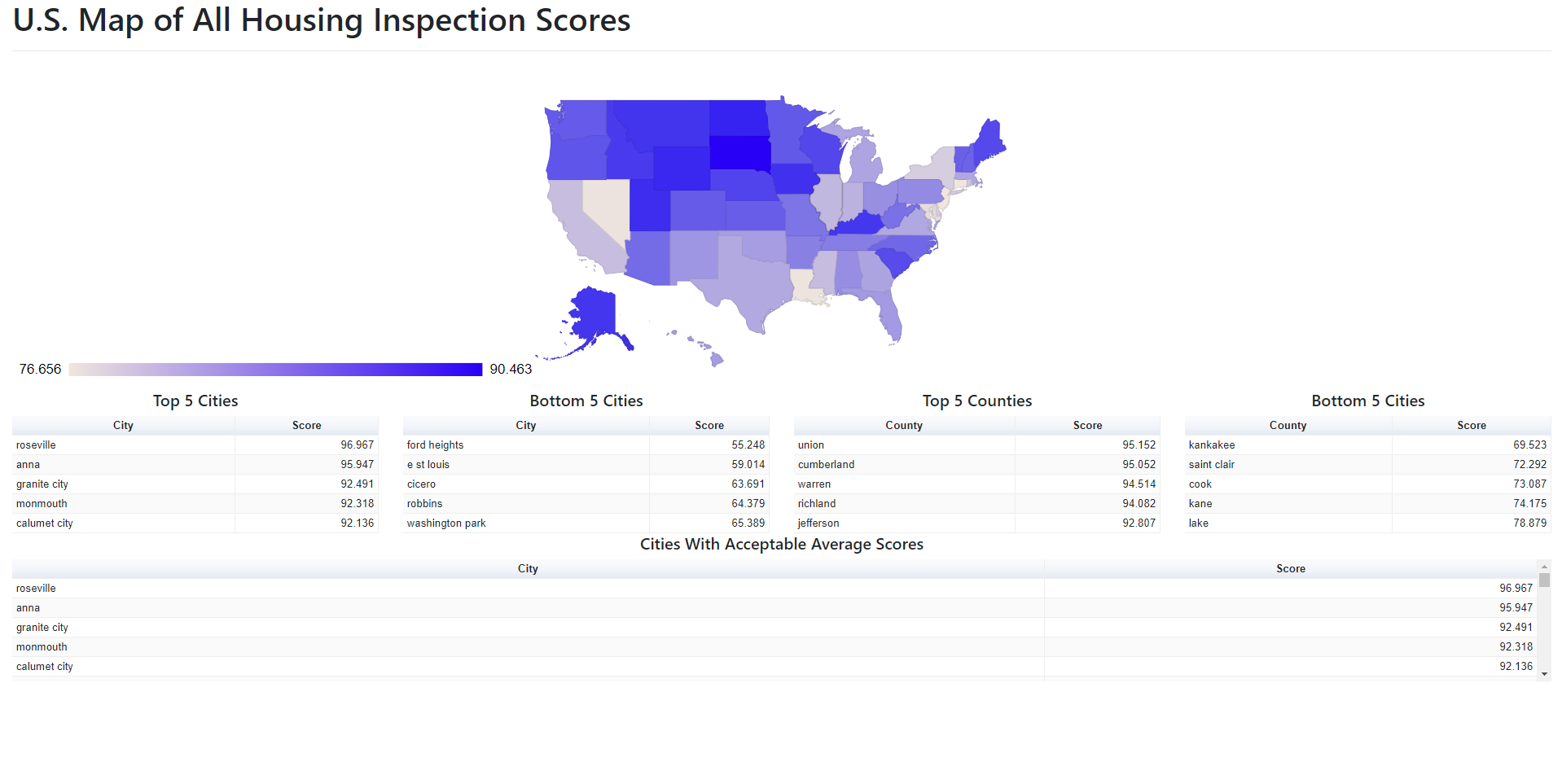
Gives top 5 States with the Overall Highest Housing Inspection Scores. The same query was used and modified to get the 5 states with the Overall Lowest Housing Inspection Scores



**5.2 Visualizations**

With the necessary SPARQL queries written, the final step was to create the HIE itself. This was accomplished by creating four separate visualizations which comprise the HIE (using the Google Charts library), where each visualization is accessible via an index page. These are discussed in greater detail below.

**Visualization: U.S. Map of All housing Inspection Scores**

*Figure 2: Geochart with selectable states for viewing city/county scores and cities with acceptable scores*

**Result**

This visualization displays the top/bottom 5 scoring cities within a state, the top/bottom 5 scoring counties within a state, and all cities with acceptable scores within a state. The user clicks on a state in the interactive map of all 50 U.S. states, and the charts below the map automatically populate with the relevant information.

**Visualization: Top States and Inspection Scores**

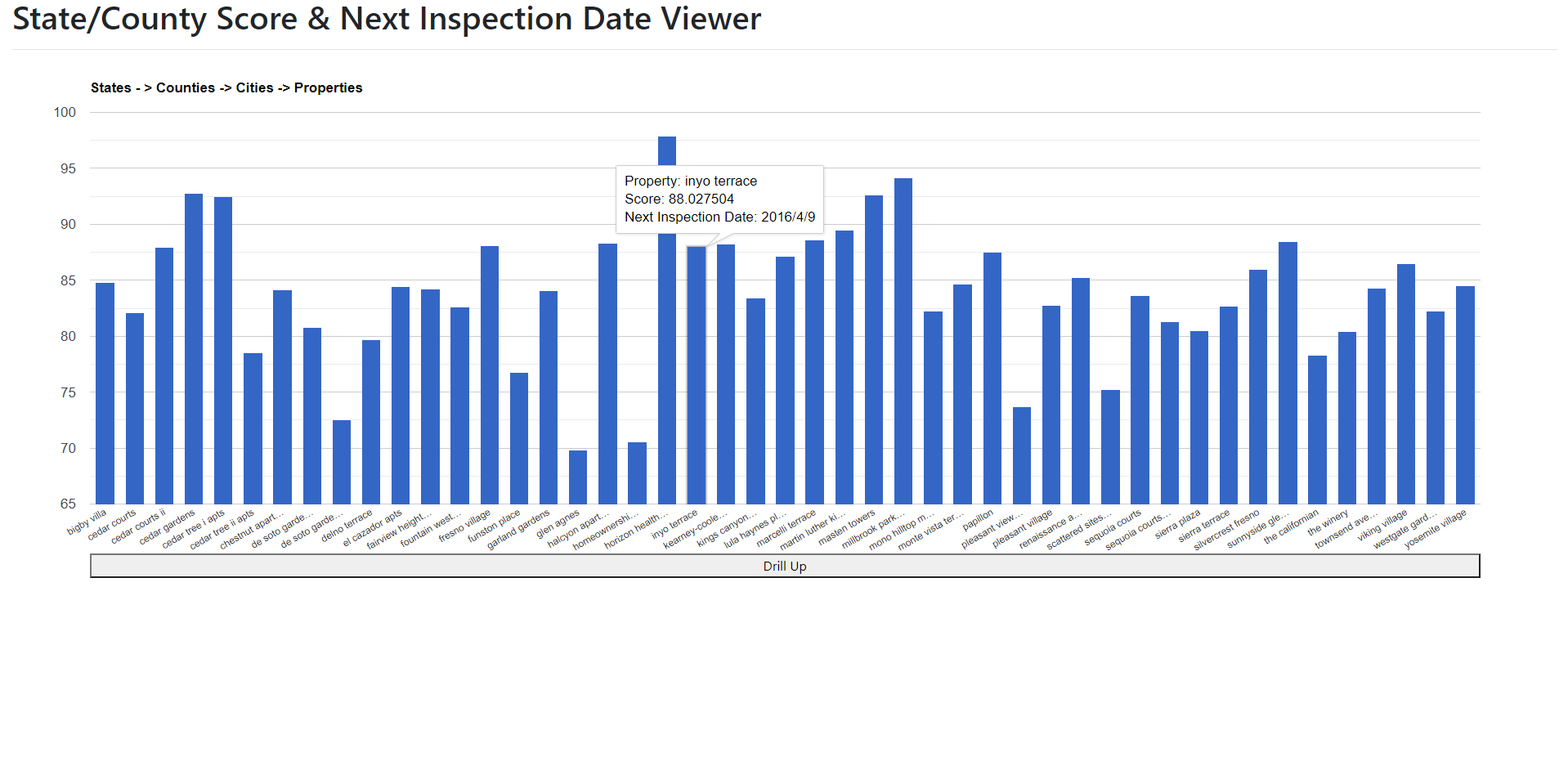


*Figure 3: Top 5 States with the Overall Highest/Lowest Housing Inspection Scores*

**Result**

This visualization displays the top/bottom 5 scoring states across the entire U.S. Hovering over each state’s corresponding bar generates a tooltip with the average score for that particular state.

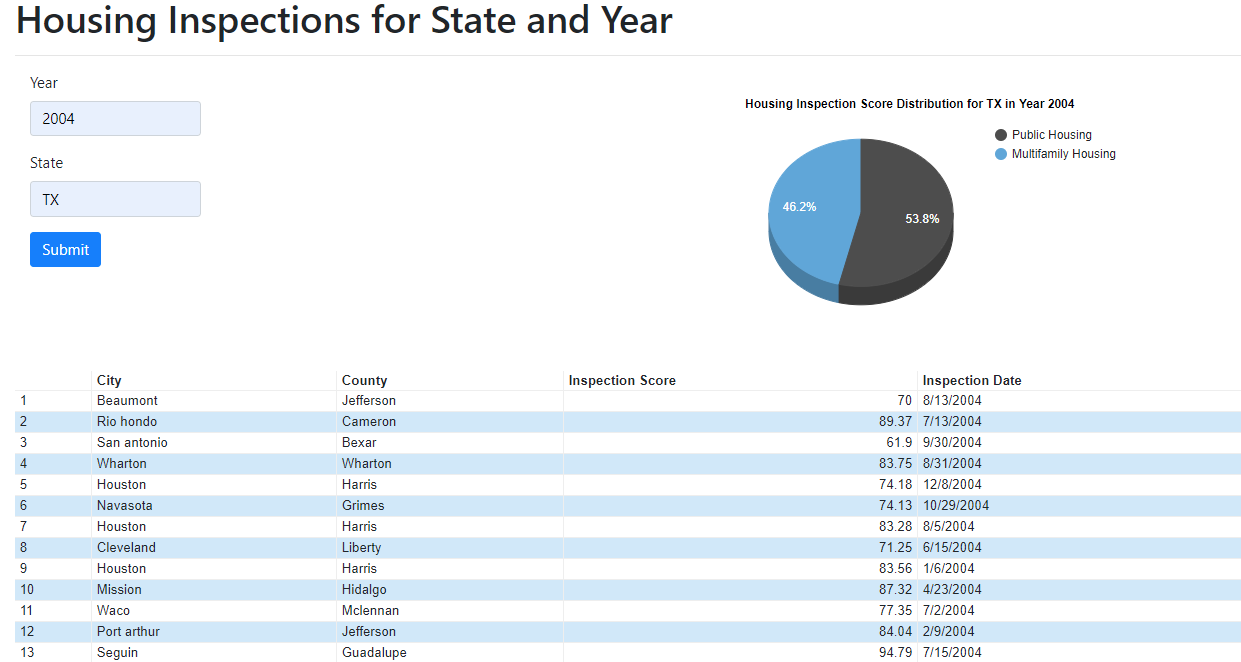
**Visualization: State/County Score & Next Inspection Date Viewer**

*Figure 4: Drilldown chart (State -> County -> City -> Property hierarchy) for scores, inspection dates*

**Result**

This visualization allows the user to view average scores at various levels within its drilldown hierarchy (e.g., state level, county level, etc.), and to drill down to individual housing properties. At this lowest level, the user may view the estimated date of next inspection for a particular housing property by hovering over its corresponding bar and viewing the date value in its tooltip.

**Visualization: Housing Inspections for State and Year**



*Figure 5: All the inspections that happened in the given year and for a given state*

**Result**

This visualization displays specific housing inspection dates for a year and state specified by the user.

**6. SUMMARY**

The initial step involved identifying the datasets that could be used for the project. We scanned the [LOGD](https://data-gov.tw.rpi.edu/wiki/Data.gov_Catalog_-_Complete) catalog and after preliminary analysis on the available datasets, we chose Datasets 1258 and 1259 for our final concept.

After this, we began analyzing our datasets in more detail to get a deeper understanding of the data and the joining attributes. SPARQL Queries were constructed and executed against Apache Jena Fuseki to get the data for the features defined in our concept.

Proceeding towards the implementation phase, we loaded our triples from the two datasets into Fuseki. We then leveraged the Google Charts library for the visualization of results, and used AJAX calls to Fuseki to populate our visualizations via the relevant SPARQL queries.

Through these steps, we were successful in implementing the HIE, which allows users to effectively view and analyze Housing Inspection Scores across the two datasets we leveraged. The only issue encountered was the need to create our own Fuseki instance when the LOGD SPARQL endpoint ceased to function, but establishing a Fuseki instance was thankfully straightforward. Thus, this project taught us the various steps involved in creating a reporting application leveraging the Semantic Web, from initial steps (identifying potential datasets), intermediate steps (establishing a Fuseki instance), and final steps (creating the actual reporting application using Google Charts).