**Ranger Motors Data Model**

**Data Dictionary**

|  |  |
| --- | --- |
| **Object** | **Description** |
| CarsForSale | This is the “main table” of our database. Each record in this table corresponds to a physical car in Ranger Motors’ inventory, the attributes of which are functionally dependent on *vin\_num* (Vehicle Identification Number), the table’s primary key. Attributes such as *year*, *mileage*, *condition* and *price* are stored in this table. The *model* attribute is a foreign key to the *Models* table. The attributes *ext\_color\_id* and *int\_color\_id* are foreign keys to the *Colors* table. |
| Models | This table contains information about each model of car. Its attributes are *model\_name* and *make*. We take the model of a car to functionally determine its make. Thus, keeping model/make information in a separate table reduced redundancy. Additionally, it mitigates the risk of erroneous data entry: any model name entered into the *CarsForSale* table must already exist in the *Models* table. A row in this table corresponds to a particular model of car (but not the year/edition of that model). Each record may correspond to many records in *CarsForSale*. |
| Colors | This table contains the names of colors in the *color­­\_name* attribute, each of which corresponds to a *color\_id*. Each record corresponds to a color and is referred to by *ext\_color\_id* and *int\_color\_id* in *CarsForSale*, to describe the color of a car’s interior or exterior. Keeping colors in a separate table reduces the risk of erroneous data entry; requiring the values of *ext\_color\_id* and *int\_color\_id* to exist in the *Colors* table prevents the existence of multiple spellings of one color, for example. |

**Important Queries Supported by the Query Interface**

The main purpose of our query interface will be to allow users to filter the records in *CarsForSale* based on attribute values. We hypothetically anticipate requests from customers interested in finding a Ranger Motors car to buy, as opposed to Ranger Motors employees. For example, users will be allowed to filter and view the results of queries, but not to manage the tables as an employee would; users cannot enter, update or delete records in the tables.

Some query examples would be to show the *make, model, year* and *price* of all cars with *mileage* greater than 100,000, all cars with automatic *transmission* and greater than 30 mpg *hwy\_fuel\_economy*. Further, these results could be sorted by *price,* or *year*. We plan to implement aggregations as well, e.g., show the average/max/count of *price* for each *make* of car. Naturally, this will require SQL statements involving WHERE, GROUP BY and HAVING clauses. Moreover, joins will be required to retrieve necessary information on makes and colors.