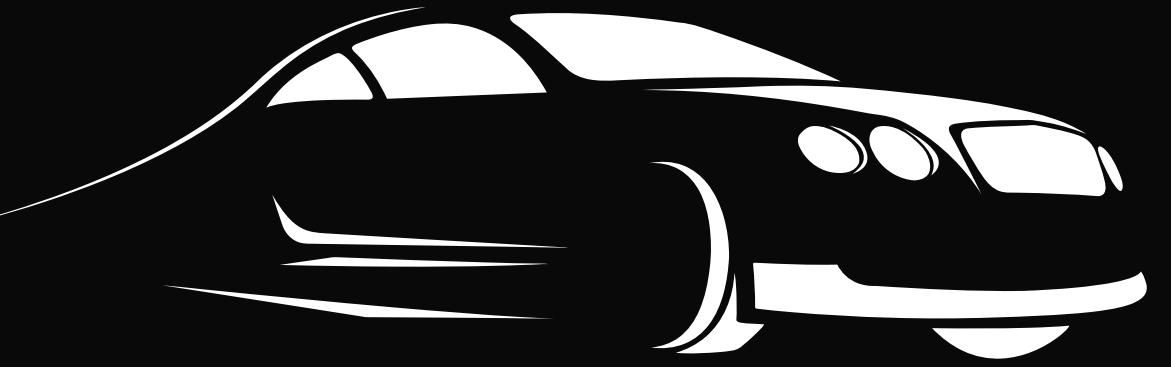


# FOX car rental

a car rental data base program by team 6



# OUR TEAM MEMBERS:

- Sriharsha Edukulla
- Joakim Eckerman
- Khush Naidu
- Vincent Nguyen

Team Number 6

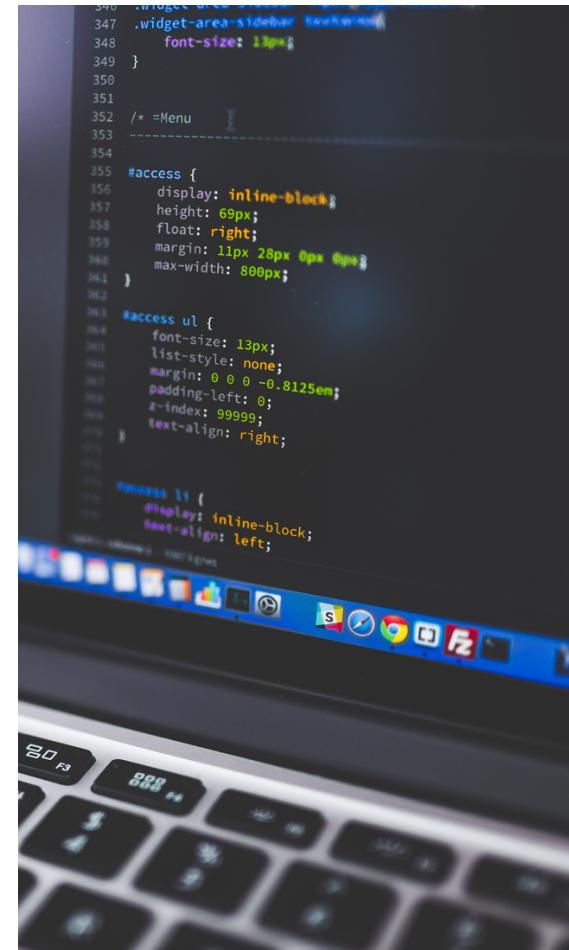




# INTRODUCTION:

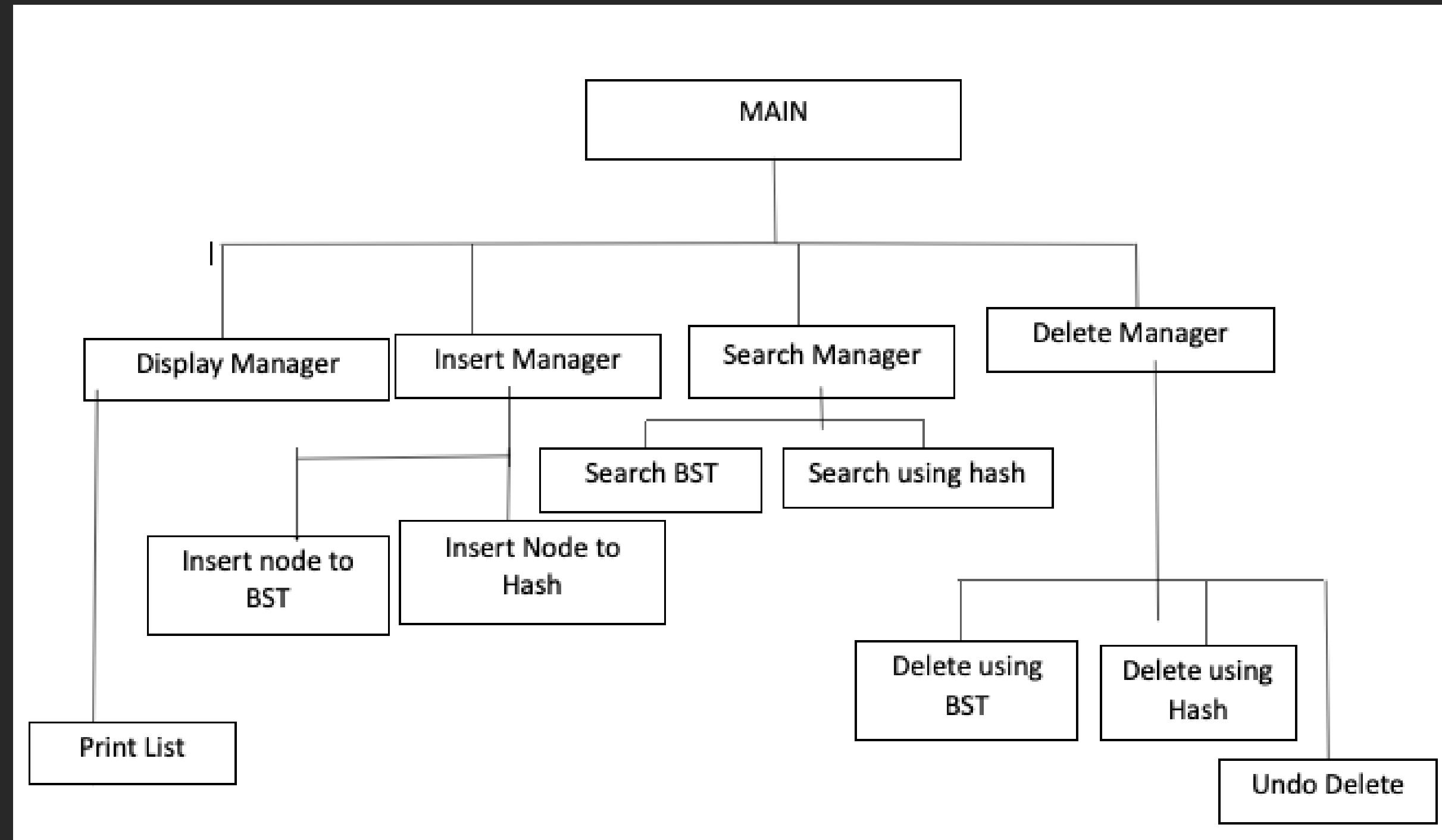
*Here at Fox Rental, our database program allows the user to view the great deals that our business has to offer. Our database allows one to search for the cars using a unique engine number as well as the name of the manufacturing company/maker.*

*It provides the option to add more cars to the catalogue as well as remove/check out cars after a deal has been done.*



# STRUCTURE CHART

Structure chart to help visualize how our program runs.



# TEAM ASSIGNMENTS

## Sriharsha Edukulla

Team Leader. Main function and integrating function calls.

---

## Vincent Nguyen

Binary Search Tree with a focus on Secondary Key.

---

## Khush Naidu

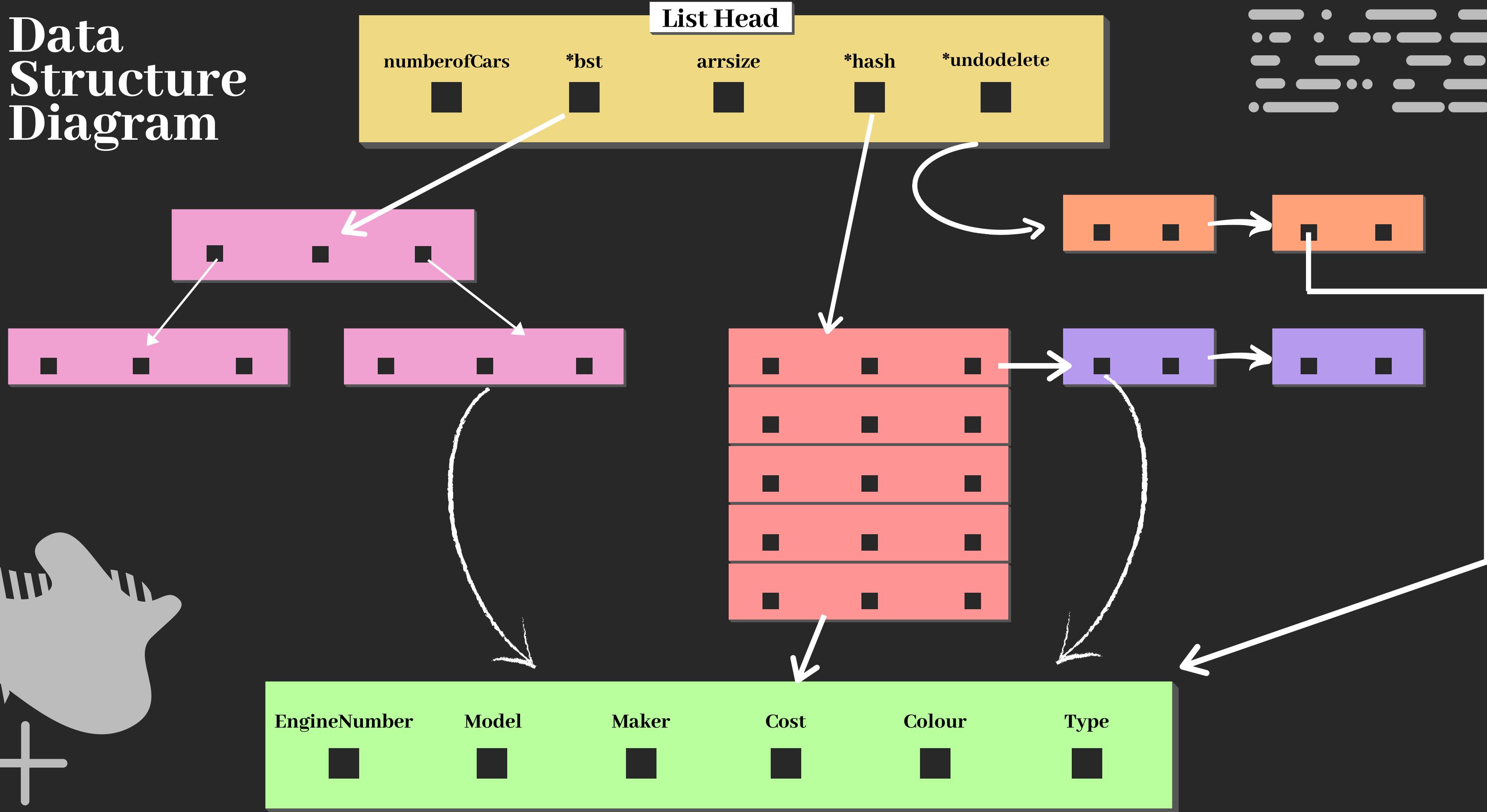
Hash Table with a focus on Primary Key as well as collisions and statistics of the table.

---

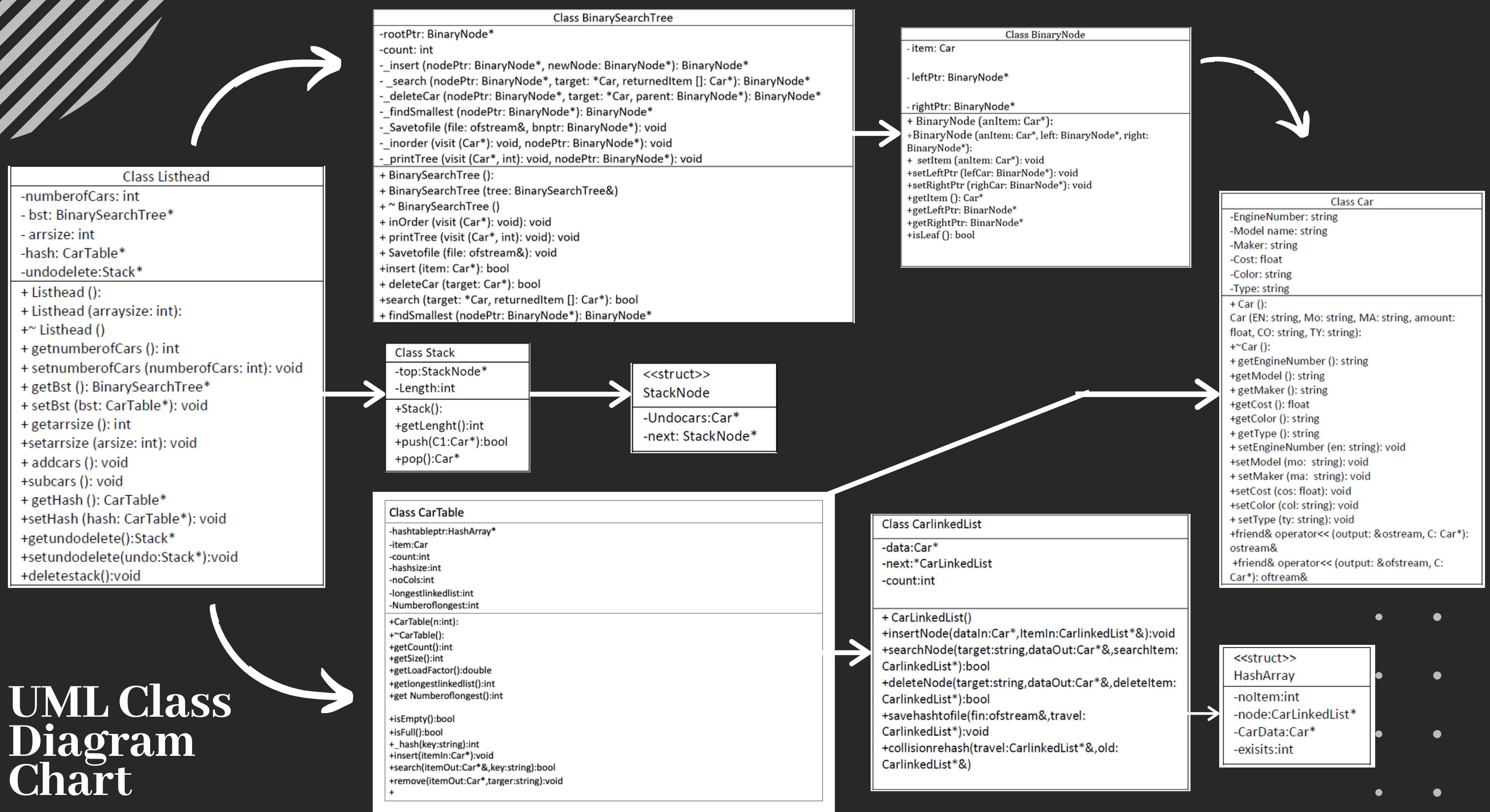
## Joakim Eckerman

Screen Output of the various managers. Made the display readable.

# Data Structure Diagram



# UML Class Diagram Chart



# HASH FUNCTION

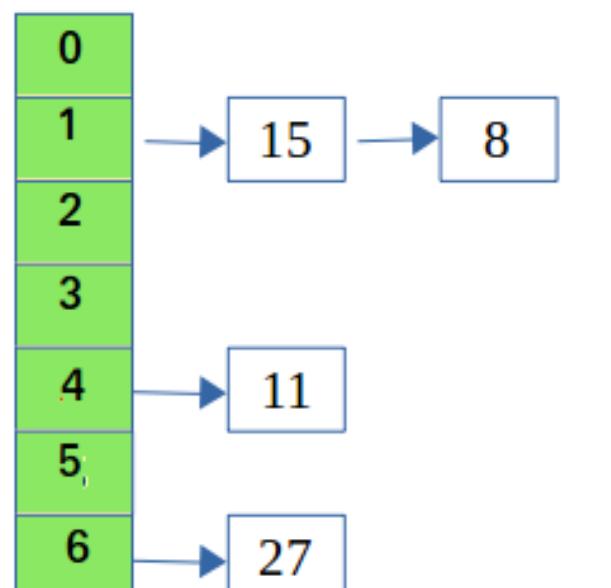
```
50
51 int CarTable::_hash(string key)
52 {
53     int sum = 0;
54     for (int i = 0; key[i]; i++)
55         sum += key[i];
56     return sum % hashSize;
57 }
```

# *How did we handle collisions in the hash table?*

*We handled the collisions using the concept of chaining. Chaining involves the use of linked lists. This enabled us to store more than value in each bucket of the hash table.*

Let's say hash table with 7 buckets (0, 1, 2, 3, 4, 5, 6)

Keys arrive in the Order (15, 11 , 27 , 8)



**EXAMPLE**

*thank you*

