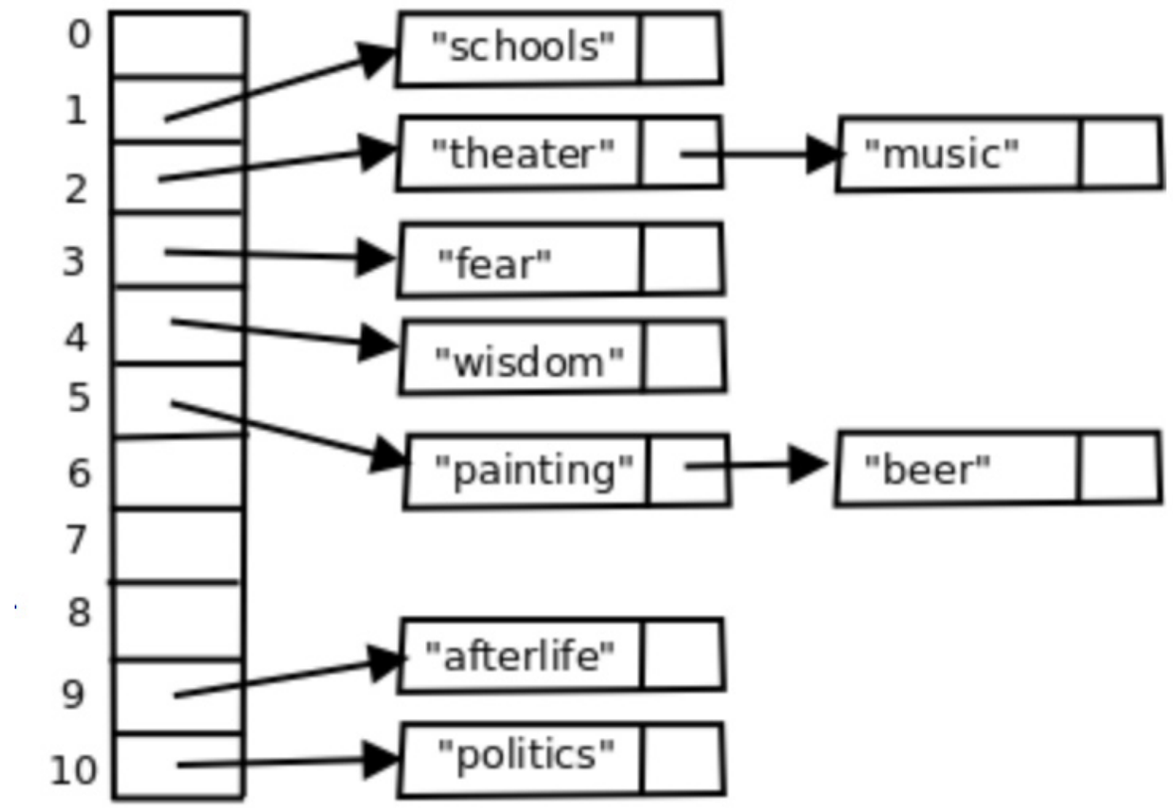
Hashset

- HashSet collection contains a group of elements of unique values stored at respective indexes.
- Full Path: System.Collections.Generic.HashSet
- Process of adding an element:
  - Generate index based on the value. Ex: index = hash code % count
  - Add the element (value) next to the linked list at the generated index.



'HashSet' collection
HashSet<T> referenceVariable = new HashSet<T>( );

- The "HashSet" class is a generic class.
- You can't set / get the element based on the key / index.
- It searches elements based on the index generated based on the search value.
- HashSet allows only one null value; Hashtable allows only one null key; but allows multiple null values.
- You can't access elements based on key / index. You can use Contains method to search for an element.
- You can't sort elements in HashSet.
- Elements must be unique; duplicate elements are not allowed.

Properties

- **Count** : Returns count of elements.

Methods

- **void Add(T value)** : Adds an element (key/value pair).
- **void Remove(T value)** : Removes an element based on specified key.
- **void RemoveWhere(Predicate)** : Remove elements that matches with condition.
- **bool Contains (T value)** : Determines whether the specified value exists.
- **void Clear()** : Removes all elements.
- **void UnionWith(IEnumerable<T>)** : Unions the hashset and specified collection.
- **void IntersectWith(IEnumerable<T>)** : Intersects the hashset and specified collection.