Java HashSet

Java HashSet

A HashSet is a collection of items where every item is unique, and it is found in the java.util package:

Example

Create a HashSet object called cars that will store strings:

```
import java.util.HashSet; // Import the HashSet class

HashSet<String> cars = new HashSet<String>();
```

Add Items

The `HashSet` class has many useful methods. For example, to add items to it, use the `add()` method:

Example

```
// Import the HashSet class
-------------------------------------------------------------------------------
import java.util.HashSet;

public class Main {
  public static void main(String[] args) {
    HashSet<String> cars = new HashSet<String>();
    cars.add("Volvo");
    cars.add("BMW");
    cars.add("Ford");
    cars.add("BMW");
    cars.add("Mazda");
    System.out.println(cars);
  }
}
-------------------------------------------------------------------------------
```

Note: In the example above, even though BMW is added twice it only appears once in the set because every item in a set has to be unique.

Check If an Item Exists

To check whether an item exists in a HashSet, use the contains() method:

Example

```
cars.contains("Mazda");
```

```
// Import the HashSet class
import java.util.HashSet;

public class Main {
  public static void main(String[] args) {
    HashSet<String> cars = new HashSet<String>();
    cars.add("Volvo");
    cars.add("BMW");
    cars.add("Ford");
    cars.add("BMW");
    cars.add("Mazda");
    System.out.println(cars.contains("Mazda"));
  }
}
```

Remove an Item

To remove an item, use the remove() method:

Example

```
cars.remove("Volvo");
```

```
// Import the HashSet class
import java.util.HashSet;

public class Main {
  public static void main(String[] args) {
    HashSet<String> cars = new HashSet<String>();
    cars.add("Volvo");
    cars.add("BMW");
    cars.add("Ford");
    cars.add("BMW");
    cars.add("Mazda");
    cars.remove("Volvo");
    System.out.println(cars);
  }
}
```

To remove all items, use the clear() method:

Example

```
cars.clear();
```

```
// Import the HashSet class
import java.util.HashSet;

public class Main {
  public static void main(String[] args) {
    HashSet<String> cars = new HashSet<String>();
    cars.add("Volvo");
    cars.add("BMW");
    cars.add("Ford");
    cars.add("BMW");
    cars.add("Mazda");
    cars.clear();
    System.out.println(cars);
  }
}
```

HashSet Size

To find out how many items there are, use the size method:

Example

```
cars.size();
// Import the HashSet class
```

```
import java.util.HashSet;

public class Main {
  public static void main(String[] args) {
    HashSet<String> cars = new HashSet<String>();
    cars.add("Volvo");
    cars.add("BMW");
    cars.add("Ford");
    cars.add("BMW");
    cars.add("Mazda");
    System.out.println(cars.size());
  }
}
```

Loop Through a HashSet

Loop through the items of an HashSet with a for-each loop:

Example

```
for (String i : cars) {
  System.out.println(i);
}

// Import the HashSet class
```

```
import java.util.HashSet;

public class Main {
  public static void main(String[] args) {
    HashSet<String> cars = new HashSet<String>();
    cars.add("Volvo");
    cars.add("BMW");
    cars.add("Ford");
    cars.add("BMW");
    cars.add("Mazda");
    for (String i : cars) {
      System.out.println(i);
    }
  }
}
```

Other Types

Items in an HashSet are actually objects. In the examples above, we created items (objects) of type "String". Remember that a String in Java is an object (not a primitive type). To use other types, such as int, you must specify an equivalent wrapper class: Integer. For other primitive types, use: Boolean for boolean, Character for char, Double for double, etc:

Example

Use a HashSet that stores Integer objects:

```
---------------------------------------------------------------------------------
import java.util.HashSet;

public class Main {
  public static void main(String[] args) {

    // Create a HashSet object called numbers
    HashSet<Integer> numbers = new HashSet<Integer>();

    // Add values to the set
    numbers.add(4);
    numbers.add(7);
    numbers.add(8);

    // Show which numbers between 1 and 10 are in the set
    for(int i = 1; i <= 10; i++) {
      if(numbers.contains(i)) {
        System.out.println(i + " was found in the set.");
      } else {
        System.out.println(i + " was not found in the set.");
      }
    }
  }
}
---------------------------------------------------------------------------------
```