```
class DataAnalyzer:
    def __init__(self):
        self.data_set = set()
        self.data_dict = {}

    def add_to_set(self, elements):
        self.data_set.update(elements)

    def remove_from_set(self, element):
        self.data_set.discard(element)

    def get_set(self):
        return self.data_set

    def create_dictionary(self, keys, values):
        if len(keys) == len(values):
            self.data_dict = dict(zip(keys, values))
        else:
            print("Error: Keys and values lists must be of equal length.")

    def update_dictionary(self, key, value):
        self.data_dict[key] = value

    def get_dictionary(self):
        return self.data_dict

    def search_dictionary(self, key):
        return key in self.data_dict

    def remove_from_dictionary(self, key):
        if key in self.data_dict:
            del self.data_dict[key]

# Example usage:
da = DataAnalyzer()
da.add_to_set([10,20,30,40])
print("Set:", da.get_set())

da.remove_from_set(3)
print("Set after removing 3:", da.get_set())

da.create_dictionary(['a', 'b', 'c'], [10, 20, 30])
print("Dictionary:", da.get_dictionary())

da.update_dictionary('d', 40)
print("Dictionary after update:", da.get_dictionary())

print("Is 'b' in dictionary?", da.search_dictionary('b'))

da.remove_from_dictionary('a')
print("Dictionary after removing 'a':", da.get_dictionary())
```

```
Set: {40, 10, 20, 30}
Set after removing 3: {40, 10, 20, 30}
Dictionary: {'a': 10, 'b': 20, 'c': 30}
Dictionary after update: {'a': 10, 'b': 20, 'c': 30, 'd': 40}
Is 'b' in dictionary? True
Dictionary after removing 'a': {'b': 20, 'c': 30, 'd': 40}
```