[Basic ABC 1](#_Toc71898916)

[Hello World ABC 1](#_Toc71898917)

[Sub 1](#_Toc71898918)

[Sub Sub 1](#_Toc71898919)

[Sub Sub Sub 1](#_Toc71898920)

# Basic CQEngine

## Hello World CQEngine

package examples.introduction.novice;

import com.googlecode.cqengine.attribute.Attribute;

import com.googlecode.cqengine.attribute.SimpleAttribute;

import com.googlecode.cqengine.query.option.QueryOptions;

public class Person {

    public final String name;

    public Person(String name) {

        this.name = name;

    }

    @Override

    public String toString() {

        return "Person{name=" + name +"}";

    }

    // -------------------------- Attributes --------------------------public static final Attribute<Person, String> NAME = new SimpleAttribute<Person, String>("name") {

        public String getValue(Person person, QueryOptions queryOptions) { return person.name; }

    };

}

package examples.introduction.novice;

import com.googlecode.cqengine.ConcurrentIndexedCollection;

import com.googlecode.cqengine.IndexedCollection;

import com.googlecode.cqengine.index.navigable.NavigableIndex;

import com.googlecode.cqengine.query.Query;

import static com.googlecode.cqengine.query.QueryFactory.\*;

/\*\*

 \*

 \*/

public class Introduction {

    public static void main(String[] args) {

        // Create an indexed collection

        IndexedCollection<Person> people = new ConcurrentIndexedCollection<Person>();

        // Add some indexes...

        people.addIndex(NavigableIndex.onAttribute(Person.NAME));

        // Add some objects to the collection...

        addData(people);

        // -------------------------- Run some queries --------------------------

        System.out.println("People whose names contain Arthur:");

        Query<Person> query1 = contains(Person.NAME, "Arthur");

        people.retrieve(query1).forEach(System.out::println);

    }

    private static void addData(IndexedCollection<Person> people) {

        people.add(new Person("Arthur White"));

        people.add(new Person("Arthur Pendragon"));

        people.add(new Person("Arthur Bach"));

        people.add(new Person("Elizabeth I"));

        people.add(new Person("James I after Elizabeth I"));

    }

}

### Sub

#### Sub Sub

##### Sub Sub Sub