Java API for JSON Processing

- javax.json

- Main Classes and Interfaces

- Creating an Object Model

- Navigating an Object Model

- Writing an Object Model

JSR 374 (JSON Processing) API

API to parse, transform, and query JSON data using the object model or the streaming model

```
<dependency>
    <groupId>org.glassfish</groupId>
    <artifactId>javax.json</artifactId>
    <version>1.0.4</version>
</dependency>
```

Main Classes and Interfaces in javax.json

Class or Interface	Description
Json	Contains static methods to create instances of JSON parsers, builders, and generators. This class also contains methods to create parser, builder, and generator factory objects.
JsonReader	Reads JSON data from a stream and creates an object model in memory.
JsonObjectBuilder JsonArrayBuilder	Create an object model or an array model in memory by adding elements from application code.
JsonWriter	Writes an object model from memory to a stream.
JsonValue	Represents an element (such as an object, an array, or a value) in JSON data.
JsonStructure	Represents an object or an array in JSON data. This interface is a subtype of JsonValue.
JsonObject JsonArray	Represent an object or an array in JSON data. These two interfaces are subtypes of JsonStructure.
JsonString JsonNumber	Represent data types for elements in JSON data. These two interfaces are subtypes of JsonValue.
JsonException	Indicates that a problem occurred during JSON processing.

Creating an Object Model

From JSON file

```
JsonReader reader = Json.createReader(new FileReader("jsondata.txt"));

JsonStructure jsonst = reader.read();
```

From Application Code

```
JsonObject model = Json.createObjectBuilder()
  .add("firstName", "Duke")
  .add("age", 28)
  .add("streetAddress", "100 Internet Dr")
  .add("phoneNumbers", Json.createArrayBuilder()
    .add(Json.createObjectBuilder()
      .add("type", "home")
      .add("number", "222-222-2222")))
  .build();
```

Navigating an Object Model

See navigateTree() example

Writing an Object Model

```java
StringWriter stWriter = new StringWriter();
try (JsonWriter jsonWriter = Json.createWriter(stWriter)) {
  jsonWriter.writeObject(model);
}

String jsonData = stWriter.toString();
System.out.println(jsonData);
```