Rollno - 31436

Assignment B7

Title - study and demonstrate the use of encoding and decoding Json object using Java.

Problem statement : study and demonstrate the use of encoding and decoding Json objects using Java.

Objective : To understand and implement encoding and decoding of Json object

S/w & H/w req :
Eclipse, Java, 64 bit os, fedora keyboard mouse.

Theory -

Json extension is bundled with PHP by default from version 5.2.0 so there is no need of any special environment.

JSON functions :

1) JSON encode : it returns the json representation of a value.

2) Json decode - It decodes a json string

3) json_last_error - It returns the last error occured

Encoding :

json - encode ( ) function is used for encoding which returns Json representation of a value.

Syntax :

```
string json.encode ($value [ , $ option= 0 ] )
```

The value parameters specifies value being specified. It works only with UTF 8 encoded data.

Decoding :

json_decode ( ) function is used for decoding Json object to PHP.

Syntax.-

```
Json-decode ( $ json [ , $ assoc = False [ ,
    $ depth = 5, ] [ , $ options = 0 ] ] } )
```

parameters :

json-string : It is encoded string which must be UTF - 8 encoded data.

assoc : It is boolean type parameter when set to true, returned objects will be converted into associated arrays.

- depth : It is an integer type parameters which specifies recursion depth.

- options : It is an integer type bitmask of Json decode . It supprerrer Json string.

Conclusion :-

We have studied and demonstreeed the use of encoding and decoding object using java.

Source Code:

```
import java.util.ArrayList;
import java.util.Scanner;

import org.bson.Document;

import org.json.JSONArray;
import org.json.JSONObject;

import com.mongodb.DB;
import com.mongodb.DBCollection;
import com.mongodb.MongoClient;
import com.mongodb.client.FindIterable;
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoDatabase;

class JsonObject
{


        public static void main(String args[])
        {

                MongoClient mongoClient = new MongoClient("localhost", 27017);
    MongoDatabase db = mongoClient.getDatabase("assignment_B7");
                MongoCollection coll = db.getCollection("student");
                FindIterable<Document> docs = coll.find();
                ArrayList<Document> arr = new ArrayList<Document>();
```

```
for(Document doc : docs)
{
    arr.add(doc);
    System.out.println(doc);
}
//Encoding
        JSONObject g_data = new JSONObject();
        g_data.put("name", "g");
        g_data.put("age", 17);
        g_data.put("div", "g");
        JSONObject h_data = new JSONObject();
        h_data.put("name", "h");
        h_data.put("age", 18);
        h_data.put("div", "h");
        JSONArray newData = new JSONArray();
        newData.put(g_data);
        newData.put(h_data);
        for(int i=0; i<newData.length(); i++)
        {
            Document doc = Document.parse(newData.get(i).toString());
            coll.insertOne(doc);
        }
        System.out.println("\nAfter insert");
        for(Document docss : docs)
        {
            System.out.println(docss);
        }
```

```
                //Decoding
                JSONObject obj2 = new JSONObject(arr.get(0));
                Student s1 = new Student();
                s1.name = (String) obj2.get("name");
                s1.age = (Double) obj2.get("age");
                s1.div = (String) obj2.get("div");
                s1.display();
    }
}
class Student {
    public String name, div;
    public double age;
    public void display() {
        System.out.println("name: " + name);
        System.out.println("age: " + age);
        System.out.println("div: " + div);
    }
}
```