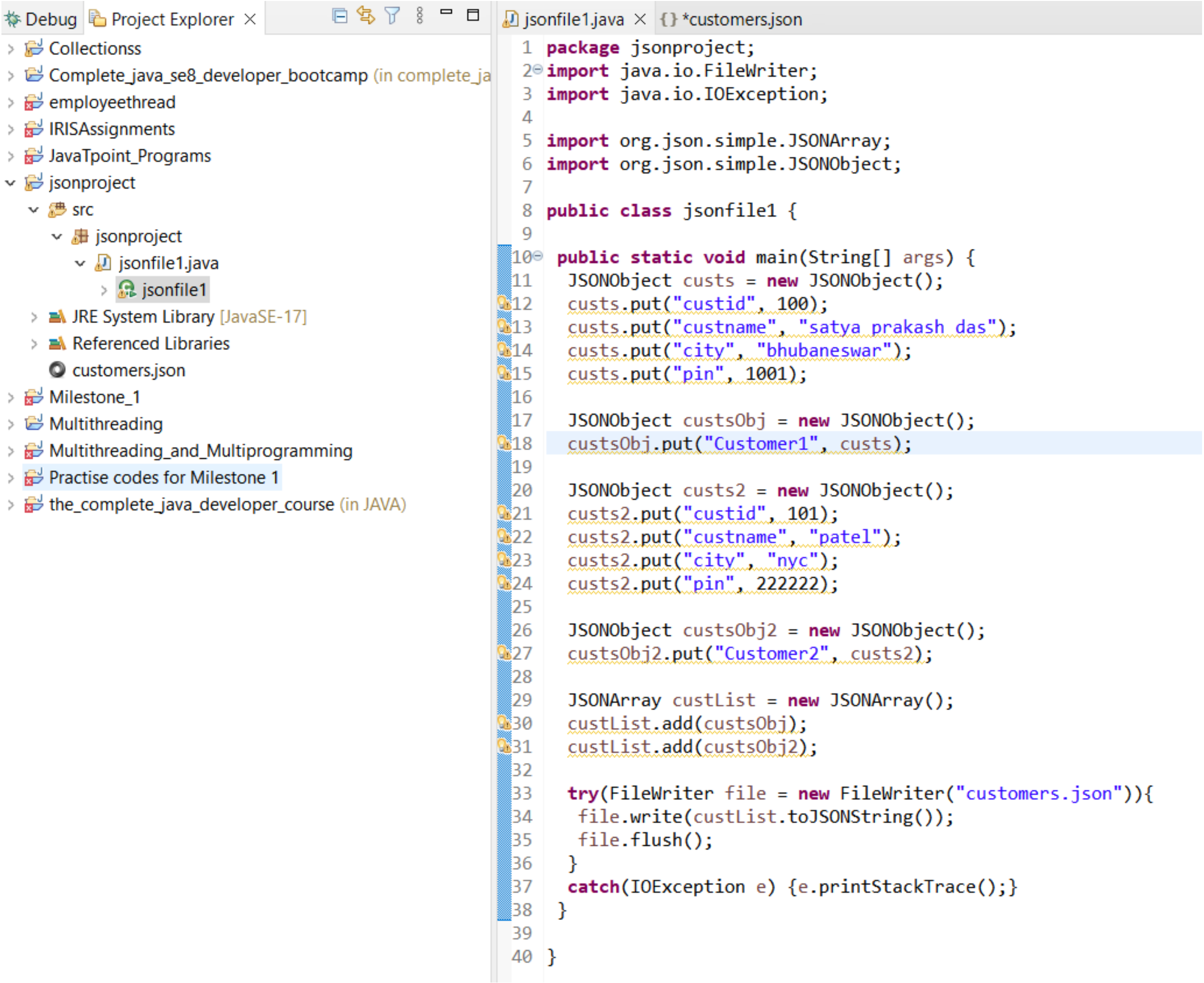
Learning Java 9 : Databases and Multithreading in Java

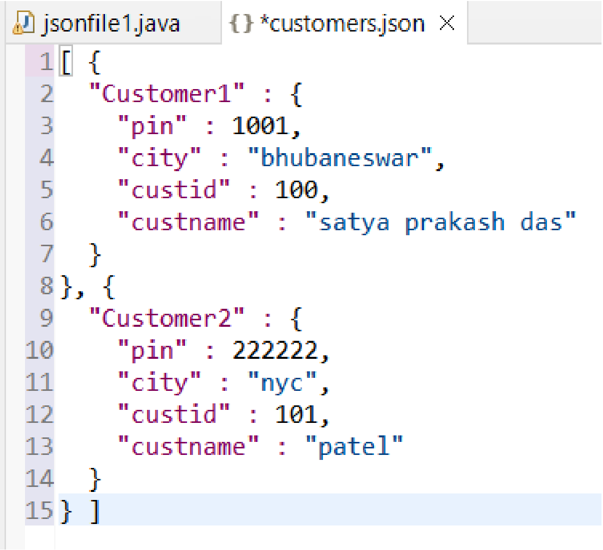
User Assignment-1

Problem Statement:

1. Create a json file called **customers.json**, with fields custId, custName, city and pin and populate with some data.
2. Write a hibernate program to read the json data from the json file and store it in the Postgre mysql database.

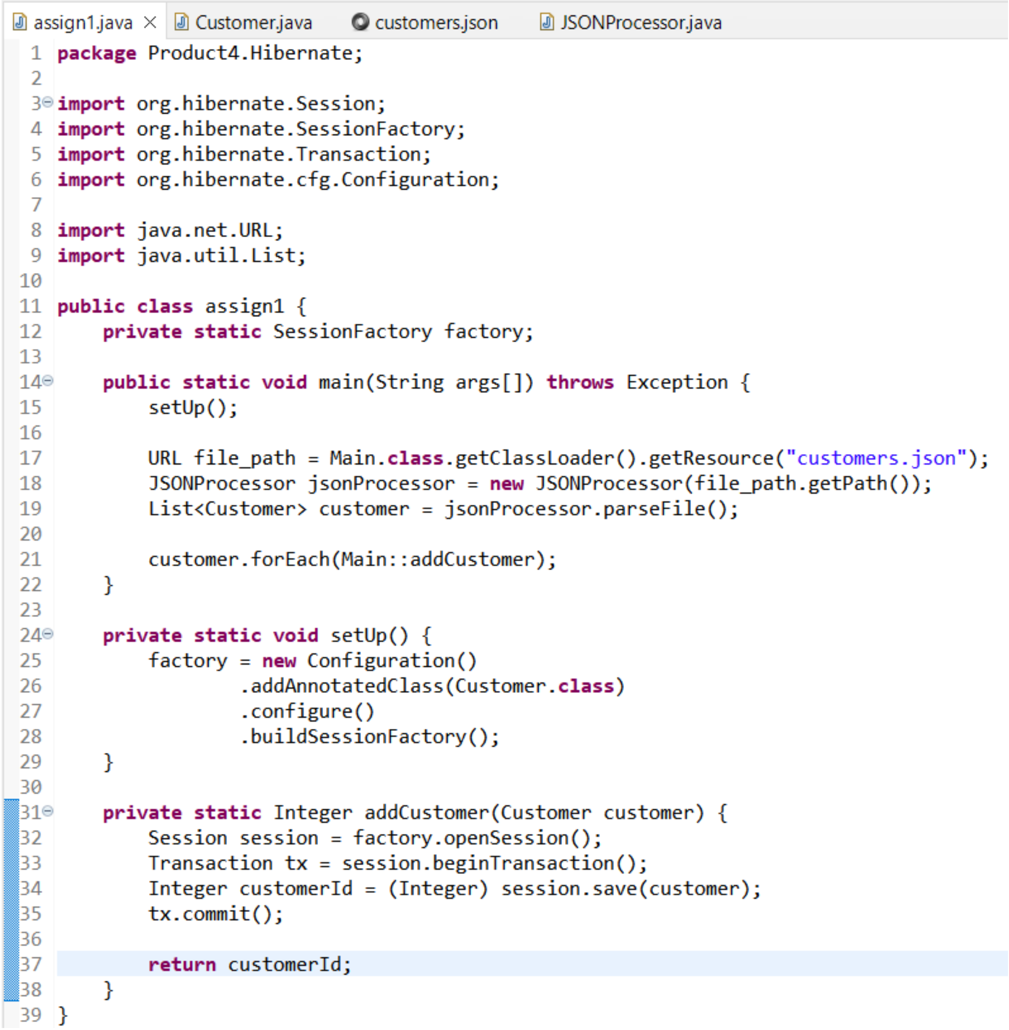


```java
package jsonproject;
import java.io.FileWriter;
import java.io.IOException;

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

public class jsonfile1 {

 public static void main(String[] args) {
  JSONObject custs = new JSONObject();
  custs.put("custid", 100);
  custs.put("custname", "satya prakash das");
  custs.put("city", "bhubaneswar");
  custs.put("pin", 1001);

  JSONObject custsObj = new JSONObject();
  custsObj.put("Customer1", custs);

  JSONObject custs2 = new JSONObject();
  custs2.put("custid", 101);
  custs2.put("custname", "patel");
  custs2.put("city", "nyc");
  custs2.put("pin", 222222);

  JSONObject custsObj2 = new JSONObject();
  custsObj2.put("Customer2", custs2);

  JSONArray custList = new JSONArray();
  custList.add(custsObj);
  custList.add(custsObj2);

  try(FileWriter file = new FileWriter("customers.json")){
   file.write(custList.toJSONString());
   file.flush();
  }
  catch(IOException e) {e.printStackTrace();}
 }

}
```

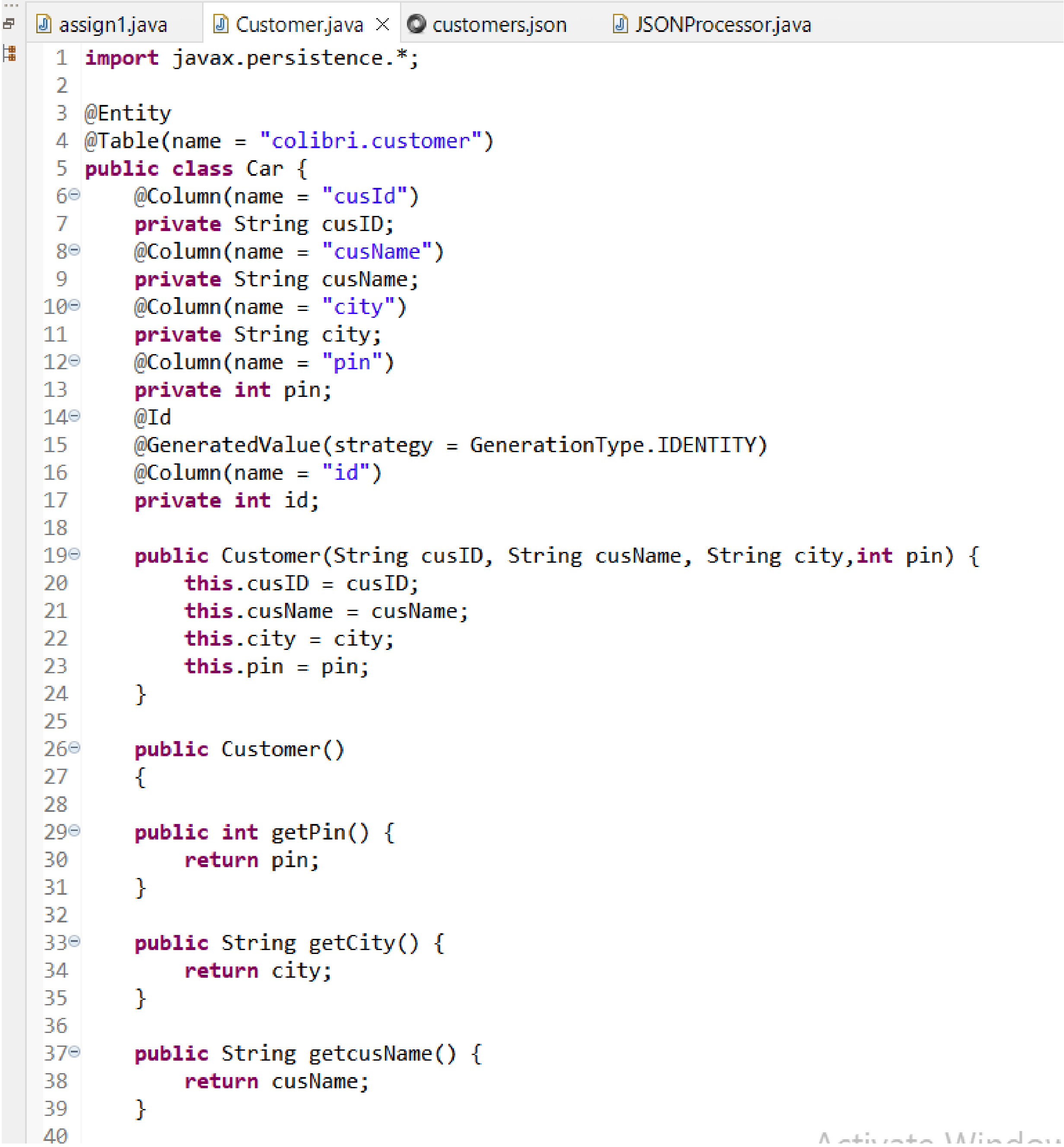


```json
[ {
  "Customer1" : {
    "pin" : 1001,
    "city" : "bhubaneswar",
    "custid" : 100,
    "custname" : "satya prakash das"
  }
}, {
  "Customer2" : {
    "pin" : 222222,
    "city" : "nyc",
    "custid" : 101,
    "custname" : "patel"
  }
} ]
```

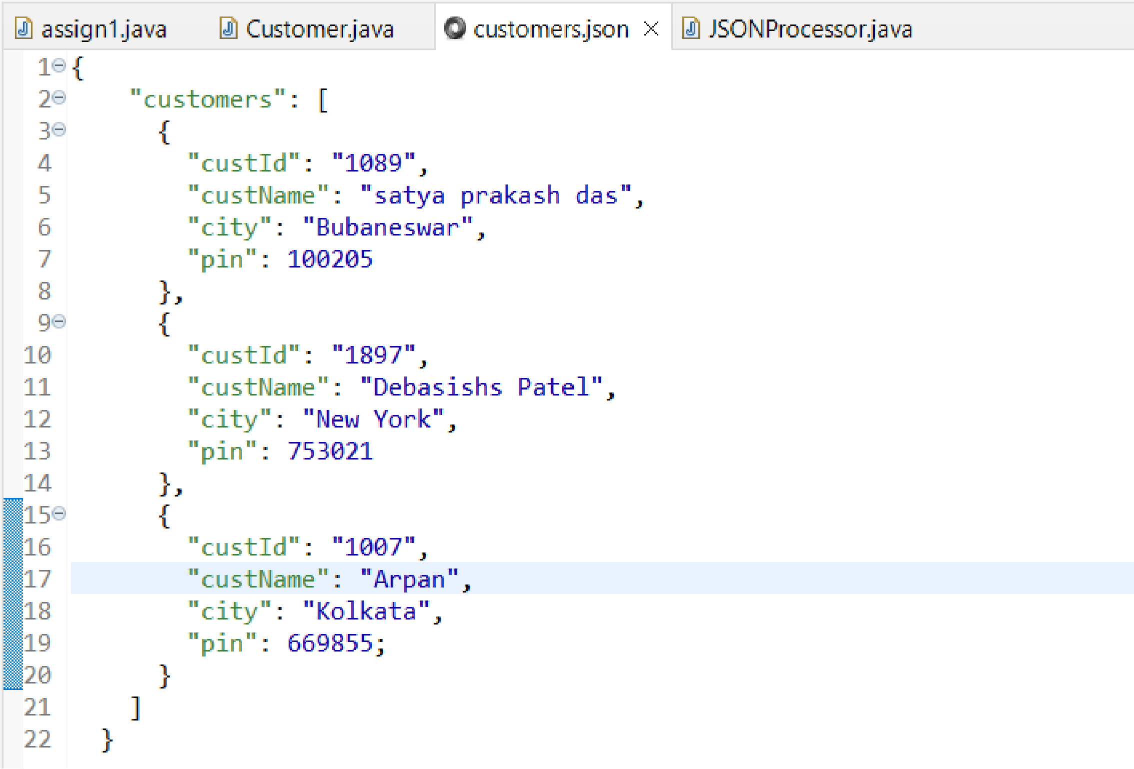
2.)



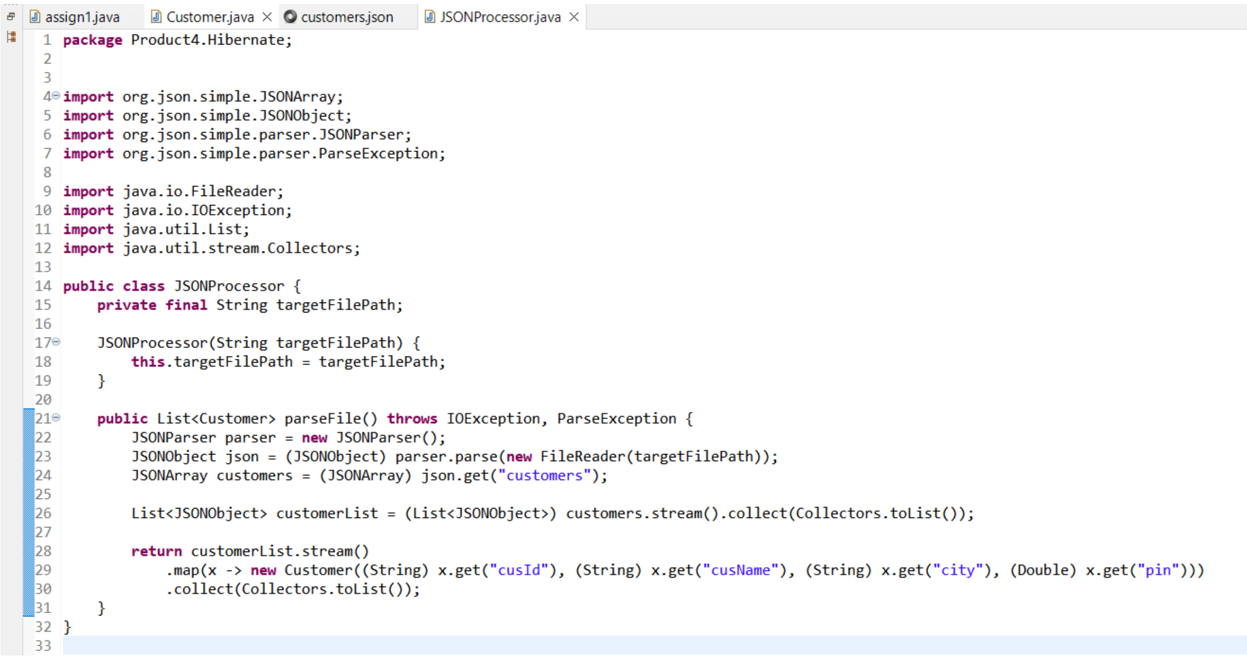
```java
package Product4.Hibernate;

import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;
import org.hibernate.cfg.Configuration;

import java.net.URL;
import java.util.List;

public class assign1 {
    private static SessionFactory factory;

    public static void main(String args[]) throws Exception {
        setUp();

        URL file_path = Main.class.getClassLoader().getResource("customers.json");
        JSONProcessor jsonProcessor = new JSONProcessor(file_path.getPath());
        List<Customer> customer = jsonProcessor.parseFile();

        customer.forEach(Main::addCustomer);
    }

    private static void setUp() {
        factory = new Configuration()
                .addAnnotatedClass(Customer.class)
                .configure()
                .buildSessionFactory();
    }

    private static Integer addCustomer(Customer customer) {
        Session session = factory.openSession();
        Transaction tx = session.beginTransaction();
        Integer customerId = (Integer) session.save(customer);
        tx.commit();

        return customerId;
    }
}
```



```java
import javax.persistence.*;

@Entity
@Table(name = "colibri.customer")
public class Car {
    @Column(name = "cusId")
    private String cusID;
    @Column(name = "cusName")
    private String cusName;
    @Column(name = "city")
    private String city;
    @Column(name = "pin")
    private int pin;
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name = "id")
    private int id;

    public Customer(String cusID, String cusName, String city,int pin) {
        this.cusID = cusID;
        this.cusName = cusName;
        this.city = city;
        this.pin = pin;
    }

    public Customer()
    {

    public int getPin() {
        return pin;
    }

    public String getCity() {
        return city;
    }

    public String getcusName() {
        return cusName;
    }

```



```
{
    "customers": [
      {
        "custId": "1089",
        "custName": "satya prakash das",
        "city": "Bubaneswar",
        "pin": 100205
      },
      {
        "custId": "1897",
        "custName": "Debasishs Patel",
        "city": "New York",
        "pin": 753021
      },
      {
        "custId": "1007",
        "custName": "Arpan",
        "city": "Kolkata",
        "pin": 669855;
      }
    ]
  }
```



```java
package Product4.Hibernate;


import org.json.simple.JSONArray;
import org.json.simple.JSONObject;
import org.json.simple.parser.JSONParser;
import org.json.simple.parser.ParseException;

import java.io.FileReader;
import java.io.IOException;
import java.util.List;
import java.util.stream.Collectors;

public class JSONProcessor {
    private final String targetFilePath;

    JSONProcessor(String targetFilePath) {
        this.targetFilePath = targetFilePath;
    }

    public List<Customer> parseFile() throws IOException, ParseException {
        JSONParser parser = new JSONParser();
        JSONObject json = (JSONObject) parser.parse(new FileReader(targetFilePath));
        JSONArray customers = (JSONArray) json.get("customers");

        List<JSONObject> customerList = (List<JSONObject>) customers.stream().collect(Collectors.toList());

        return customerList.stream()
            .map(x -> new Customer((String) x.get("cusId"), (String) x.get("cusName"), (String) x.get("city"), (Double) x.get("pin")))
            .collect(Collectors.toList());
    }
}
```