SL- VI
Expt. 6

Aim:
Design Persistent Objects using JDO and implement min 10 queries on objects using JDOQL in ObjectDB NOSQL DATABASE - ***Cassandra***

Steps:
First install and configure ***Cassandra*** from,
https://sl6it.wordpress.com/2015/12/10/4-installation-of-nosql-database-cassandra/

```
# Start cassandra daemon
~/cassandra/bin/cassandra -f
```

# The cassandra daemon should start in the foreground
# (**don't press ctrl + c**; as it'll terminate the daemon)

```
# In another terminal
~/cassandra/bin/cqlsh
```

# Create keyspace

```
cqlsh> CREATE KEYSPACE keyspace1
WITH REPLICATION = { 'class' : 'SimpleStrategy', 'replication_factor' : 1 };

cqlsh> USE keyspace1;
```

# Open Eclipse Editor

# Install DataNucleus plugin
In Eclipse-> Go to Help -> Install new software -> Add-> Set name as **DataNucleus** **->** Set Location as, **http://www.datanucleus.org/downloads/eclipse-update/** **->**ok

After searching the plugin on internet, it will display one checkbox ***"DataNucleus Eclipse plugin"*** -> Check the checkbox->Next->Next->I accept->Finish-> Ignore the warning (if any) -> Restart eclipse -> yes

# In eclipse ->Create new java project -> **exp6**

# Download **DataNucleus AccessPlatform** from

http://sourceforge.net/projects/datanucleus/files/datanucleus-accessplatform/4.2.3/datanucleus-accessplatform-cassandra-4.2.3.zip/download

# Extract the downloaded zip file
When you open the zip you will find DataNucleus jars in the ***lib*** directory, and dependency jars in the ***deps*** directory.

# From the ***lib*** directory copy all jar files ***except*** following

datanucleus-jodatime-4.1.1.jar
datanucleus-java8-4.2.1.jar
datanucleus-guava-4.1.3.jar

# In eclipse -> right click on project **exp6** -> Paste
# One by one, right click on pasted jarfile -> Build path -> add to build path
# Add all pasted jar files to build path

# Similarly from the ***deps*** folder copy ***all*** jar files

# Once again In eclipse -> right click on project **exp6** -> Paste
# One by one, right click on pasted jarfile -> Build path -> add to build path
# Add all pasted jar files to build path

# In eclipse -> Create a new class - > **Product ->** add following code in it

```
package exp6;
import javax.jdo.annotations.IdGeneratorStrategy;
import javax.jdo.annotations.PersistenceCapable;
import javax.jdo.annotations.Persistent;
import javax.jdo.annotations.PrimaryKey;

@PersistenceCapable
public class Product
{
    @PrimaryKey
    @Persistent(valueStrategy=IdGeneratorStrategy.INCREMENT)
    long id;
    String name = null;
    String description = null;
    double price = 0.0;

    public Product(String name, String desc, double price)
    {
        this.name = name;
        this.description = desc;
        this.price = price;
    }
}
```

# save the file

# In eclipse -> Create a new class - > **Book ->** add following code in it

```
package exp6;
import javax.jdo.annotations.PersistenceCapable;

@PersistenceCapable
public class Book extends Product
{   String author=null;
    String isbn=null;
    String publisher=null;

    public Book(String name, String desc, double price, String author,
                String isbn, String publisher)
    {   super(name,desc,price);
        this.author = author;
        this.isbn = isbn;
        this.publisher = publisher;
    }
}
```

# save the file

# In eclipse -> Create a new class - > **Inventory ->** add following code in it

```
package exp6;
import java.util.HashSet;
import java.util.Set;

import javax.jdo.annotations.PersistenceCapable;
import javax.jdo.annotations.PrimaryKey;
@PersistenceCapable
public class Inventory
{ @PrimaryKey
    String name = null;
    @SuppressWarnings({ "rawtypes", "unchecked" })
      Set<Product> products = new HashSet();

    public Inventory(String name)
    {
        this.name = name;
    }

    public Set<Product> getProducts() {return products;}
}
```

# save the file

# Plugin configuration

Right click on **exp6 project**-> Properties-> **DataNucleus** -> check "**Enable project specific settings**" ->**Add JARS-** > add all **18** jar files (**10** from **lib** folder, **8** from **deps** folder)-> apply -> ok

# Plugin configuration – Enhancer

Right click on **exp6 project**-> Properties -> double click on DataNucleus -> **Enhancer** -> check "**Enable project specific settings**" ->check **verbose mode** -> check **Capture Output ->** Give **persistence-unit name** -> **TEST**

# Plugin configuration – SchemaTool

Right click on **exp6 project**-> Properties ->Under DataNucleus -> **SchemaTool**-> check "**Enable project specific settings**" ->check **verbose mode ->** Give **persistence-unit name** -> **TEST**

# Right-click **project exp6** -> select **DataNucleus**->"**Add DataNucleus Support**"

# Right-click on ***package*** **exp6** -> **DataNucleus**-> select "**Create JDO XML Metadata File**" -> Give Filename-> **package.jdo** -> finish

# Right-click on ***package*** **exp6** -> **DataNucleus**-> select "**Create persistence.xml file**"

# Edit **persistence.xml** file

```
<?xml version="1.0" encoding="UTF-8" ?>
<persistence xmlns="http://java.sun.com/xml/ns/persistence"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://java.sun.com/xml/ns/persistence
http://java.sun.com/xml/ns/persistence/persistence_1_0.xsd" version="1.0">

    <persistence-unit name="TEST">
        <mapping-file>/home/s/workspace/exp6/src/exp6/package.jdo</mapping-file>

         <properties>
            <property name="javax.jdo.option.ConnectionURL" value="cassandra:"/>
            <property name="javax.jdo.mapping.Schema" value="keyspace1"/>
            <property name="datanucleus.schema.autoCreateAll" value="true"/>
         </properties>

    </persistence-unit>


</persistence>
```

# Save the file

# Download the ***cassandra-java driver*** from
http://downloads.datastax.com/java-driver/cassandra-java-driver-2.0.2.tar.gz

# Extract the downloaded file.

# Paste all jar files (total 12) from the extracted folder into your eclipse project
(Note :: 10 jar files are present under lib folder)

# In eclipse -> One by one, right click on pasted jarfile -> Build path -> add to build path
# Add all pasted jar files to the build path

# In eclipse -> Create new class- > **MyApp** -> add following code in it

```
package exp6;
import java.util.Iterator;
import java.util.List;
import javax.jdo.JDOHelper;
import javax.jdo.PersistenceManager;
import javax.jdo.PersistenceManagerFactory;
import javax.jdo.Query;
import javax.jdo.Transaction;

public class MyApp {
      public static void main(String[] args)
      {//TEST is the persistent unit name
            PersistenceManagerFactory pmf = 
JDOHelper.getPersistenceManagerFactory("TEST");
            PersistenceManager pm = pmf.getPersistenceManager();
            
            Transaction tx=pm.currentTransaction();
            try
            {
                tx.begin();
                Inventory inv = new Inventory("My Inventory");
                Product product = new Product("Sony Discman", "A standard 
discman from Sony", 49.99);
                Product product1 = new Product("Sony xperia z1", "A smart 
phone", 149.99);
                inv.getProducts().add(product);
                inv.getProducts().add(product1);
                pm.makePersistent(inv);

                Query q = pm.newQuery("SELECT FROM " + Product.class.getName() 
+" WHERE price < 150.00 ORDER BY price ASC");

            // add some more JDOQL queries here

                List<Product> products = (List<Product>)q.execute();
                Iterator<Product> iter = products.iterator();
                while (iter.hasNext())
                  {
                        Product p = iter.next();

                        System.out.println("Name: "+p.name+"\t Description: 
"+p.description+"\tPrice: "+p.price);
                  }
          tx.commit();
            }
            finally
            {
                if (tx.isActive())
                {
                    tx.rollback();
                }
                pm.close();
            }
      }

}
```

# Save the file

# Right-click on ***project* exp6** -> DataNucleus-> select "**Run Enhancer tool**"

# Output

```
ENHANCED (Persistable) : exp6.Product
ENHANCED (Persistable) : exp6.Inventory
ENHANCED (Persistable) : exp6.Book
DataNucleus Enhancer completed with success for 3 classes.
```

# Make sure that ***cassandra*** is running before doing next step

# Right-click on ***project* exp6** -> DataNucleus-> select "**Run Schema tool**" -> Set Connection URL -> **cassandra:** -> next -> finish

# Output

```
DataNucleus SchemaTool : Input Files
>>  /home/s/workspace1/exp6/bin/exp6/Book.class
>>  /home/s/workspace1/exp6/bin/exp6/Inventory.class
>>  /home/s/workspace1/exp6/bin/exp6/Product.class
>>  /home/s/workspace1/exp6/bin/exp6/MyApp.class
>>  /home/s/workspace1/exp6/bin/exp6/package.jdo
>>  /home/s/workspace1/exp6/src/exp6/package.jdo
DataNucleus SchemaTool completed successfully
```

# Right click on **MyApp.java** -> Run as -> Java Application

# Output

```
log4j:WARN No appenders could be found for logger (DataNucleus.General).
log4j:WARN Please initialize the log4j system properly.
Name: Sony Discman    Description: A standard discman from Sony   Price: 49.99
Name: Sony xperia z1  Description: A smart phone      Price: 149.99
```

# In cqlsh prompt, verify the output

cqlsh> **SELECT * FROM** Product;

# Output

```
 id | description                  | name           | price
----+------------------------------+----------------+--------
 41 | A standard discman from Sony |   Sony Discman |  49.99
 42 |  A smart phone               |   Sony xperia z1 | 149.99
```

# Now edit ***MyApp.java*** to add some more ***JDOQL queries*** one by one (**minimum 10** are expected)

# Then, Run ***MyApp.java***

# Verify the output in cqlsh prompt by using **SELECT** query.

# A very good tutorial on ***JDOQL queries*** is available in following link,

https://cloud.google.com/appengine/docs/java/datastore/jdo/queries

References:

http://www.datanucleus.org/products/datanucleus/jdo/guides/eclipse.html
http://www.datanucleus.org/products/datanucleus/jdo/samples/tutorial_cassandra.html
http://www.datanucleus.org/products/datanucleus/datastores/cassandra.html
https://cloud.google.com/appengine/docs/java/datastore/jdo/queries

--
Prof. S. T. Kolhe
(IT DEPT, SRES COE Kopargaon)