1. **Design a Java application to retrieve various Book details (bookId, bookTitle, bookAuthor, bookPublishYear etc.) from a Library Database created using MySQL RDBMS. Make use of Remote Object communication concept**.

**Code:**

**Book.java**

package roiDB;

import java.io.Serializable;

public class Book implements Serializable {

private static final long serialVersionUID = 1L;

private int bookID;

private String bookTitle;

private String bookAuthor;

private int bookPublishYear;

public int getBookID() {

return bookID;

}

public void setBookID(int bookID) {

this.bookID = bookID;

}

public String getBookTitle() {

return bookTitle;

}

public void setBookTitle(String bookTitle) {

this.bookTitle = bookTitle;

}

public String getBookAuthor() {

return bookAuthor;

}

public void setBookAuthor(String bookAuthor) {

this.bookAuthor = bookAuthor;

}

public int getBookPublishYear() {

return bookPublishYear;

}

public void setBookPublishYear(int bookPublishYear) {

this.bookPublishYear = bookPublishYear;

}

}

**IBook.java**

package roiDB;

import java.rmi.Remote;

import java.rmi.RemoteException;

import java.util.List;

public interface IBook extends Remote {

public List<Book> getBookDetails() throws RemoteException;

}

**IBookImplementer.java**

package roiDB;

import java.rmi.RemoteException;

import java.sql.\*;

import java.util.ArrayList;

import java.util.List;

public class IBookImplementer implements IBook {

static final String JDBC\_DRIVER = "com.mysql.jdbc.Driver";

static final String DB\_URL = "jdbc:mysql://localhost:3306/books";

static Connection conn = null;

static Statement stmt = null;

List<Book> bookObjects = new ArrayList<Book>();

@Override

public List<Book> getBookDetails() throws RemoteException {

try {

Class.forName(JDBC\_DRIVER);

System.out.println("Connecting to a selected database...");

conn = DriverManager.getConnection(DB\_URL, "root", "");

System.out.println("Connected database successfully...");

stmt = conn.createStatement();

String query = "SELECT \* FROM bookdetails";

ResultSet results = stmt.executeQuery(query);

while(results.next()) {

int ID = results.getInt("bookID");

String title = results.getString("bookTitle");

String author = results.getString("bookAuthor");

int publishYear = results.getInt("bookPublishYear");

Book b = new Book();

b.setBookID(ID);

b.setBookTitle(title);

b.setBookAuthor(author);

b.setBookPublishYear(publishYear);

bookObjects.add(b);

}

System.out.println("Data fetched successfully");

} catch (SQLException e) {

e.printStackTrace();

} catch (ClassNotFoundException e) {

e.printStackTrace();

}

return bookObjects;

}

}

**BookServer.java**

package roiDB;

import java.rmi.AlreadyBoundException;

import java.rmi.RemoteException;

import java.rmi.registry.LocateRegistry;

import java.rmi.registry.Registry;

import java.rmi.server.UnicastRemoteObject;

public class BookServer{

public static void main(String[] args) throws RemoteException, AlreadyBoundException {

IBookImplementer cObj= new IBookImplementer();

IBook intfC;

try {

intfC = (IBook) UnicastRemoteObject.exportObject(cObj, 0);

Registry r1 = LocateRegistry.createRegistry(1234);

r1.bind("BookService",intfC);

} catch (RemoteException e) {

// TODO Auto-generated catch block

e.printStackTrace();

} catch (AlreadyBoundException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

System.out.println("Book Server is ready");

}

}

**BookClient.java**

package roiDB;

import java.rmi.NotBoundException;

import java.rmi.RemoteException;

import java.rmi.registry.LocateRegistry;

import java.rmi.registry.Registry;

import java.util.List;

public class BookClient {

public static void main(String[] args) {

try {

Registry r1 = LocateRegistry.getRegistry(1234);

IBook ic= (IBook) r1.lookup("BookService");

System.out.println("Service accessed");

List<Book> clist = ic.getBookDetails();

for(Book c : clist)

{

System.out.println("Book Id:" + c.getBookID());

System.out.println("Book Name:" + c.getBookTitle());

System.out.println("Book Author:" + c.getBookAuthor());

System.out.println("Book Publish Year:" + c.getBookPublishYear());

}

} catch (RemoteException e) {

// TODO Auto-generated catch block

e.printStackTrace();

} catch (NotBoundException e) {

// TODO Auto-generated catch block

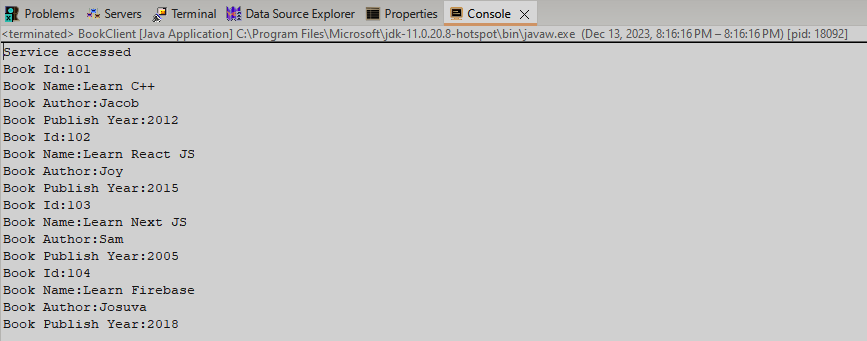
e.printStackTrace();

}

}

}

**Output:**



1. **Design a Java application to retrieve Telephone Bill Information of a customer (consumerName, billDate, billAmount, lastDate) from a telephone Bill Database created using MySQl RDBMS. Make use of Remote Object communication concept.**

**Code:**

**TelephoneBill.java**

package roiDBTelephoneBill;

import java.io.Serializable;

import java.sql.Date;

public class TelephoneBill implements Serializable {

private static final long serialVersionUID = 1L;

private String consumerName;

private Date billDate;

private int billAmount;

private Date lastDate;

public String getConsumerName() {

return consumerName;

}

public void setConsumerName(String consumerName) {

this.consumerName = consumerName;

}

public Date getBillDate() {

return billDate;

}

public void setBillDate(Date billDate) {

this.billDate = billDate;

}

public int getBillAmount() {

return billAmount;

}

public void setBillAmount(int billAmount) {

this.billAmount = billAmount;

}

public Date getLastDate() {

return lastDate;

}

public void setLastDate(Date lastDate) {

this.lastDate = lastDate;

}

public static long getSerialversionuid() {

return serialVersionUID;

}

}

ITelephoneBill.java

package roiDBTelephoneBill;

import java.rmi.Remote;

import java.rmi.RemoteException;

import java.util.List;

public interface ITelephoneBill extends Remote {

public List<TelephoneBill> getTelephoneBillDetails() throws RemoteException;

}

**ITelephoneBillImplementer.java**

package roiDBTelephoneBill;

import java.rmi.RemoteException;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.sql.Date;

import java.util.List;

public class ITelephoneBillImplementer implements ITelephoneBill {

static final String JDBC\_DRIVER = "com.mysql.jdbc.Driver";

static final String DB\_URL = "jdbc:mysql://localhost:3306/telephone";

static Connection conn = null;

static Statement stmt = null;

List<TelephoneBill> bookObjects = new ArrayList<TelephoneBill>();

@Override

public List<TelephoneBill> getTelephoneBillDetails() throws RemoteException {

try {

Class.forName(JDBC\_DRIVER);

System.out.println("Connecting to a selected database...");

conn = DriverManager.getConnection(DB\_URL, "root", "");

System.out.println("Connected database successfully...");

stmt = conn.createStatement();

String query = "SELECT \* FROM telephonebilldetails";

ResultSet results = stmt.executeQuery(query);

while(results.next()) {

String name = results.getString("consumerName");

Date date = results.getDate("billDate");

int amount = results.getInt("billAmount");

Date lastDate = results.getDate("lastDate");

TelephoneBill t = new TelephoneBill();

t.setConsumerName(name);

t.setBillDate(date);

t.setBillAmount(amount);

t.setLastDate(lastDate);

bookObjects.add(t);

}

System.out.println("Data fetched successfully");

} catch (SQLException e) {

e.printStackTrace();

} catch (ClassNotFoundException e) {

e.printStackTrace();

}

return bookObjects;

}

}

**TelephoneBillServer.java**

package roiDBTelephoneBill;

import java.rmi.AlreadyBoundException;

import java.rmi.RemoteException;

import java.rmi.registry.LocateRegistry;

import java.rmi.registry.Registry;

import java.rmi.server.UnicastRemoteObject;

public class telephoneBillServer {

public static void main(String[] args) throws RemoteException, AlreadyBoundException {

ITelephoneBillImplementer cObj= new ITelephoneBillImplementer();

try {

ITelephoneBill intfC = (ITelephoneBill) UnicastRemoteObject.exportObject(cObj, 0);

Registry r1 = LocateRegistry.createRegistry(1234);

r1.bind("TelephoneService",intfC);

} catch (RemoteException e) {

// TODO Auto-generated catch block

e.printStackTrace();

} catch (AlreadyBoundException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

System.out.println("Telephone Server is ready");

}

}

**TelephoneBillClient.java**

package roiDBTelephoneBill;

import java.rmi.NotBoundException;

import java.rmi.RemoteException;

import java.rmi.registry.LocateRegistry;

import java.rmi.registry.Registry;

import java.util.List;

public class TelephoneBillClient {

public static void main(String[] args) {

try {

Registry r1 = LocateRegistry.getRegistry(1234);

ITelephoneBill ic= (ITelephoneBill) r1.lookup("TelephoneService");

System.out.println("Service accessed");

List<TelephoneBill> clist = ic.getTelephoneBillDetails();

for(TelephoneBill c : clist)

{

System.out.println("Consumer Name:" + c.getConsumerName());

System.out.println("Bill Date:" + c.getBillDate());

System.out.println("Bill Amount:" + c.getBillAmount());

System.out.println("Last Date:" + c.getLastDate());

}

} catch (RemoteException e) {

// TODO Auto-generated catch block

e.printStackTrace();

} catch (NotBoundException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

**Output:**

