CS 4513
Fall 2016
Dr. Le Gruenwald
Riley Fant
112936608
riley.fant@ou.edu
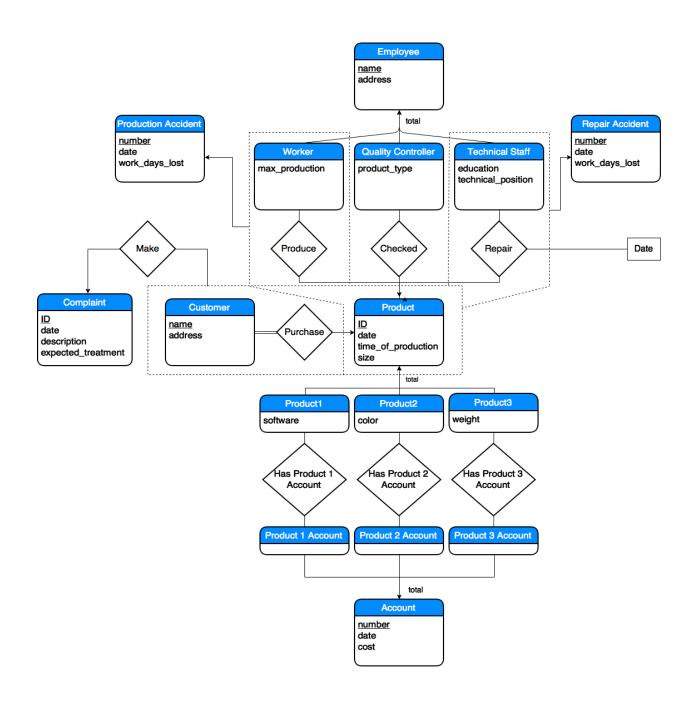Future Inc. Database System

Task 1

1.1 ER Diagram

1.2 Relational Database Schema

employee(name, address)

worker(employee.name, production_limit)
quality_controller(employee.name, product_type)
technical_staff(employee.name, education, technical_position)
customer(name, address)
product(pid, production_date, production_time, size, worker.name, qc.name, ts.name, repair_date, customer.name)
product1(pid, software)
product2(pid, color)
product3(pid, weight)
product1_account(pid, account.number)
product2_account(pid, account.number)
product3_account(pid, account.number)
account(number, pid, date_est, cost)
complaint(cid, pid, customer.name, date, description, expected_treatment)
accident(number, employee.name, product.pid, date, work_days_lost)

Task 2

Table	Table Attributes			
	Attribute Name	Type	Size (bytes)	Constraints
Employee	name	varchar(32)	32	Must be unique, non null
	address	varchar(64)	64	
Worker	employee.name	varchar(32)	32	Must Exist as an Employee.name
	production_limit	integer	4	positive
QualityController	employee.name	varchar(32)	32	Must Exist as an Employee.name
	product_type	integer	32	domain = {1,2,3}
TechnicalStaff	employee.name	varchar(32)	32	Must Exist as an Employee.name
	education	varchar(4)	4	non null
	technical_position	varchar(16)	16	non null
Product	pid	integer	4	Must be unique
	production_date	date	4	
	production_time_minutes	integer	4	positive
	size	varchar(8)	8	
	w.name	varchar(32)	32	Must not be null. Must Exist as a Worker.Employee.name
	qc.name	varchar(32)	32	Must not be null. Must exist as a QualityController.Employee.name
	ts.name	varchar(32)	32	Must exist as a TechnicallStaff.Employee.name
	repair_date	date	4	
	customer.name	varchar(32)	32	must exist as customer.name
Product1	pid	integer	4	Must exist as a Product.pid
	software	varchar(128)	128	
	account.number	integer	4	must exist as account.number
Product2	pid	integer	32	Must exist as a Product.pid
	color	varchar(16)	16	
	account.number	integer	4	must exist as account.number
Product3	pid	integer	4	Must exist as a Product.pid
	weight	integer	4	

	account.number	integer	4	must exist as account.number
	number	integer	4	Must be unique, non null
	date_est	date	4	
	cost	integer	4	positive
Account	pid	integer	4	must exist as product.pid
	number	integer	4	Must be unique, non null
	date	date	4	
	days_lost	integer	4	
	product.pid	integer	4	Must not be null. Must exist as a Product.pid
Accident	employee.name	varchar(32)	32	Must not be null. Must exist as an Employee.name
	name	varchar(32)	32	Must be unique, non null
Customer	address	varchar(64)	64	
	complaint_id	integer	4	Must be unique, non null
	cdate	date	4	
	description	varchar(512)	512	
	treatment	varchar(16)	16	
	customer.name	varchar(32)	32	Must not be null. Must exist as a Customer.name
Complaint	product.pid	integer	4	Must not be null. Must exist as a Product.pid

Task 3

3.1

EMPLOYEE
- o NAME – Hash Table because it is a String and a primary key

WORKER
- o EMPLOYEE.NAME – Hash Table because it is a String and a primary key

QUALITYCONTROLLER
- o EMPLOYEE.NAME – Hash Table because it is a String and a primary key

TECHNICALSTAFF
- o EMPLOYEE.NAME – Hash Table because it is a String and a primary key

PRODUCT
- o PID – A B+ Tree because it is an int and a primary key
- o W.NAME – Hash Table because it is a string and a foreign key
- o QC.NAME – Hash Table because it is a string and a foreign key
- o TS.NAME – Hash Table because it is a string and a foreign key

PRODUCT1
- o PID – A B+ Tree because it is an int and a primary/foreign key

PRODUCT2
- o PID – A B+ Tree because it is an int and a primary/foreign key

PRODUCT3
- o PID – A B+ Tree because it is an int and a primary/foreign key

ACCOUNT
- o NUMBER – A B+ Tree because it is an int and a primary key

ACCIDENT
- o NUMBER – A B+ Tree because it is an int and a primary key
- o PRODUCT.PID – A B+ tree because it is an int and a foreign key
- o EMPLOYEE.NAME – A Hash Table because it is a String and a foreign key

CUSTOMER
- o NAME – A Hash Table because it is a String and a foreign key

COMPLAINT
- o CID – A B+ tree because it is an int and a primary key
- o CUSTOMER.NAME – A Hash Table because it is a String and a foreign key
- o PRODUCT.PID – A B+ tree because it is an int and a foreign key

3.2

Indexes will be added on all foreign keys to facilitate faster queries on those attributes.

Task 4

```
CREATE TABLE "EMPLOYEE" (
  "NAME" VARCHAR2(32) PRIMARY KEY USING INDEX ENABLE,
  "ADDRESS" VARCHAR2(64)
);

CREATE TABLE "WORKER" (
  "EMPLOYEE.NAME" VARCHAR2(32) PRIMARY KEY USING INDEX ENABLE,
  "PRODUCTION_LIMIT" INTEGER,
  FOREIGN KEY ("EMPLOYEE.NAME") REFERENCES EMPLOYEE("NAME"),
  CHECK ("PRODUCTION_LIMIT">0)
);

CREATE TABLE "QUALITYCONTROLLER" (
  "EMPLOYEE.NAME" VARCHAR2(32) PRIMARY KEY USING INDEX ENABLE,
  "PRODUCT_TYPE" INTEGER,
  FOREIGN KEY ("EMPLOYEE.NAME") REFERENCES EMPLOYEE("NAME"),
  CONSTRAINT "CHK_RODUCT_TYPE" CHECK ("PRODUCT_TYPE" IN (1, 2, 3))
);

CREATE TABLE "TECHNICALSTAFF" (
  "EMPLOYEE.NAME" VARCHAR2(32) PRIMARY KEY USING INDEX ENABLE,
  "EDUCATION" VARCHAR2(4) NOT NULL,
  "TECHNICAL_POSITION" VARCHAR2(16) NOT NULL,
  FOREIGN KEY ("EMPLOYEE.NAME") REFERENCES EMPLOYEE("NAME")
);

CREATE TABLE "CUSTOMER" (
  "NAME" VARCHAR2(32) PRIMARY KEY USING INDEX ENABLE,
  "ADDRESS" VARCHAR2(32)
);


CREATE TABLE "PRODUCT" (
  "PID" INTEGER PRIMARY KEY USING INDEX ENABLE,
  "PRODUCTION_DATE" DATE,
  "PRODUCTION_TIME_MINUTES" INTEGER,
  "SIZE" VARCHAR2(8),
  "W.NAME" VARCHAR2(32),
  "QC.NAME" VARCHAR2(32),
  "TS.NAME" VARCHAR2(32),
  "REPAIR_DATE" DATE,
  "CUSTOMER.NAME" VARCHAR2(32),
  CHECK ("PRODUCTION_TIME_MINUTES">0),
```

```
  FOREIGN KEY ("W.NAME") REFERENCES "WORKER"("EMPLOYEE.NAME"),
  FOREIGN KEY ("QC.NAME") REFERENCES
"QUALITYCONTROLLER"("EMPLOYEE.NAME")
);

CREATE TABLE "ACCOUNT" (
  "NUMBER" INTEGER PRIMARY KEY USING INDEX ENABLE,
  "DATE_EST" DATE,
  "COST" INTEGER,
  "PRODUCT.PID" INTEGER,
  CHECK ("COST">0),
  FOREIGN KEY ("PRODUCT.PID") REFERENCES "PRODUCT"("PID")
);

CREATE TABLE "PRODUCT1" (
  "PID" INTEGER PRIMARY KEY USING INDEX ENABLE,
  "SOFTWARE" VARCHAR2(128),
  "ACCOUNT.NUMBER" INTEGER,
  FOREIGN KEY ("PID") REFERENCES "PRODUCT"("PID"),
  FOREIGN KEY ("ACCOUNT.NUMBER") REFERENCES "ACCOUNT"("NUMBER")
);

CREATE TABLE "PRODUCT2" (
  "PID" INTEGER PRIMARY KEY USING INDEX ENABLE,
  "COLOR" VARCHAR2(16),
  "ACCOUNT.NUMBER" INTEGER,
  FOREIGN KEY ("PID") REFERENCES "PRODUCT"("PID"),
  FOREIGN KEY ("ACCOUNT.NUMBER") REFERENCES "ACCOUNT"("NUMBER")
);

CREATE TABLE "PRODUCT3" (
  "PID" INTEGER PRIMARY KEY USING INDEX ENABLE,
  "WEIGHT" INTEGER,
  "ACCOUNT.NUMBER" INTEGER,
  FOREIGN KEY ("PID") REFERENCES "PRODUCT"("PID"),
  FOREIGN KEY ("ACCOUNT.NUMBER") REFERENCES "ACCOUNT"("NUMBER")
);



CREATE TABLE "ACCIDENT" (
  "NUMBER" INTEGER PRIMARY KEY USING INDEX ENABLE,
  "DATE" DATE,
```

```
  "DAYS_LOST" INTEGER,
  "PRODUCT.PID" INTEGER,
  "EMPLOYEE.NAME" VARCHAR2(32),
  FOREIGN KEY ("PRODUCT.PID") REFERENCES "PRODUCT"("PID"),
  FOREIGN KEY ("EMPLOYEE.NAME") REFERENCES "EMPLOYEE"("NAME")
);

CREATE TABLE "COMPLAINT" (
  "CID" INTEGER PRIMARY KEY USING INDEX ENABLE,
  "CDATE" DATE,
  "DESCRIPTION" VARCHAR2(512),
  "TREATMENT" VARCHAR2(16),
  "CUSTOMER.NAME" VARCHAR2(32),
  "PRODUCT.PID" INTEGER,
  FOREIGN KEY ("PRODUCT.PID") REFERENCES "PRODUCT"("PID"),
  FOREIGN KEY ("CUSTOMER.NAME") REFERENCES "CUSTOMER"("NAME")
);
```

Table "EMPLOYEE" created.

Table "WORKER" created.

Table "QUALITYCONTROLLER" created.

Table "TECHNICALSTAFF" created.

Table "CUSTOMER" created.

Table "PRODUCT" created.

Table "ACCOUNT" created.

Table "PRODUCT1" created.

Table "PRODUCT2" created.

Table "PRODUCT3" created.

Table "ACCIDENT" created.

Table "COMPLAINT" created.

Task 5

```java
import java.io.BufferedReader;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;
import java.io.PrintWriter;
import java.io.UnsupportedEncodingException;
import java.sql.CallableStatement;
import java.sql.Connection;
import java.sql.Date;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.Scanner;

public class FutureDBDriver {

    @SuppressWarnings("deprecation")
    public static void main(String[] args) {
        //load the oracle driver
        try {
            Class.forName("oracle.jdbc.OracleDriver");
        }
        catch(Exception x) {
            System.out.println( "Unable to load the driver class!"
);
        }

        //connect to the database
        Connection dbConnection = null;
        Statement stmt = null;
        try{

dbConnection=DriverManager.getConnection("jdbc:oracle:thin:@//oracle.c
s.ou.edu:1521/pdborcl.cs.ou.edu","fant6608","RBpx4Ac3");
            stmt = dbConnection.createStatement();
        }catch( SQLException x ){
            System.out.println( "Couldn't get connection!" );
            System.exit(0);
        }
```

```java
        //entry point
        boolean continu = true;
        while(continu) {

                //user enters query
                Scanner scanner = new Scanner(System.in);
                int input = start(scanner);

                switch(input) {
                case 1:   //Enter a new employee (2/month).
                        //Get input
                        System.out.println("Enter the new employee's
name:");
                        String name = scanner.nextLine();
                        System.out.println("Enter the new employee's
address:");
                        String address = scanner.nextLine();

                        //Execute Query
                        try {
                                CallableStatement query1 =
dbConnection.prepareCall("{CALL INSERT_NEW_EMPLOYEE(?,?)}");
                                query1.setString(1, name);
                                query1.setString(2, address);
                                query1.executeUpdate();
                        } catch (SQLException e) {
                                System.out.println("Failed to insert new
employee " + name + ".");
                                e.printStackTrace();
                                break;
                        }

                        System.out.println("Enter the type of employee
(Worker, Technical Staff, or Quality Controller:");
                        String employeeType = scanner.nextLine();

                        if (employeeType.equalsIgnoreCase("Worker")) {
                                System.out.println("Enter the production
limit for " + name + ":");
                                int productionLimit = scanner.nextInt();
                                try {
                                        String sql = "INSERT INTO
WORKER(\"EMPLOYEE.NAME\", PRODUCTION_LIMIT) VALUES(?,?)";
```

```java
                                    PreparedStatement query =
dbConnection.prepareCall(sql);
                                    query.setString(1, name);
                                    query.setInt(2, productionLimit);
                                    query.executeUpdate();
                                    System.out.println("Successfully
inserted new worker " + name + ".");
                            } catch(SQLException e) {
                                    System.out.println("Failed to insert
new worker " + name + ".");
                                    e.printStackTrace();
                            }

                    }
                    else if (employeeType.equalsIgnoreCase("Quality
Controller")) {
                            System.out.println("Enter the product type
for " + name + " (1, 2, or 3):");
                            int productType = scanner.nextInt();
                            try {
                                    String sql = "INSERT INTO
QUALITYCONTROLLER(\"EMPLOYEE.NAME\", PRODUCT_TYPE) VALUES(?,?)";
                                    PreparedStatement query =
dbConnection.prepareCall(sql);
                                    query.setString(1, name);
                                    query.setInt(2, productType);
                                    query.executeUpdate();
                                    System.out.println("Successfully
inserted new quality controller " + name + ".");
                            } catch(SQLException e) {
                                    System.out.println("Failed to insert
new quality controller " + name + ".");
                                    e.printStackTrace();
                            }

                    }
                    else if (employeeType.equalsIgnoreCase("Technical
Staff")) {
                            System.out.println("Enter " + name + "'s
highest level of education (BS, MS, or PhD):");
                            String education = scanner.nextLine();
                            System.out.println("Enter " + name + "'s
technical position:");
```

```java
                            String technicalPosition =
scanner.nextLine();
                        try {
                            String sql = "INSERT INTO
TECHNICALSTAFF(\"EMPLOYEE.NAME\", EDUCATION, TECHNICAL_POSITION)
VALUES(?,?,?)";
                            PreparedStatement query =
dbConnection.prepareCall(sql);
                            query.setString(1, name);
                            query.setString(2, education);
                            query.setString(3, technicalPosition);
                            query.executeUpdate();
                            System.out.println("Successfully
inserted new technical staff " + name + ".");
                        } catch(SQLException e) {
                            System.out.println("Failed to insert
new technical staff " + name + ".");
                            e.printStackTrace();
                        }

                    }

                    break;
                case 2: /*Enter a new product associated with the
person who made the product, repaired the product if
                    it is repaired, or checked the product
(400/day).*/
                    //Get input
                    System.out.println("Enter a pid:");
                    int pid = Integer.parseInt(scanner.nextLine());
                    System.out.println("Enter the production year
(int):");
                    int year = Integer.parseInt(scanner.nextLine());
                    System.out.println("Enter the production month
(int):");
                    int month = Integer.parseInt(scanner.nextLine());
                    System.out.println("Enter the production day
(int):");
                    int day = Integer.parseInt(scanner.nextLine());
                    Date productionDate = new Date(year, month, day);
                    System.out.println("Enter a production time
(minutes):");
```

```java
                        int productionTimeMinutes =
Integer.parseInt(scanner.nextLine());
                        System.out.println("Enter a size:");
                        String size = scanner.nextLine();
                        System.out.println("Enter a worker name:");
                        String workerName = scanner.nextLine();
                        System.out.println("Enter a quality controller
name:");
                        String qcName = scanner.nextLine();
                        System.out.println("Enter a technical staff
name:");
                        String tsName = scanner.nextLine();
                        Date repairDate = null;
                        if (!tsName.equals("")) {
                            System.out.println("Enter the repair year
(int):");
                            year =
Integer.parseInt(scanner.nextLine());
                            System.out.println("Enter the repair month
(int):");
                            month =
Integer.parseInt(scanner.nextLine());
                            System.out.println("Enter the repair day
(int):");
                            day = Integer.parseInt(scanner.nextLine());
                            repairDate = new Date(year, month, day);
                        }
                        //Execute Query
                        try {
                            CallableStatement query1 =
dbConnection.prepareCall("{CALL
INSERT_NEW_PRODUCT(?,?,?,?,?,?,?,?,?)}");
                            query1.setInt(1, pid);
                            query1.setDate(2, productionDate);
                            query1.setInt(3, productionTimeMinutes);
                            query1.setString(4, size);
                            query1.setString(5, workerName);
                            query1.setString(6, qcName);
                            query1.setString(7, tsName);
                            query1.setDate(8, repairDate);
                            query1.setString(9, "");
                            query1.executeUpdate();
                        } catch (SQLException e) {
```

```java
                // TODO Auto-generated catch block
                e.printStackTrace();
            }

            System.out.println("Enter the product type (1, 2, or 3)");
            int productType = Integer.parseInt(scanner.nextLine());
            switch (productType) {
            case 1:
                System.out.println("Enter the name of the software name for product 1:");
                String softwareName = scanner.nextLine();
                try {
                    CallableStatement query1 = dbConnection.prepareCall("INSERT INTO PRODUCT1(PID, SOFTWARE) VALUES (?,?)");
                    query1.setInt(1, pid);
                    query1.setString(2, softwareName);
                    query1.executeUpdate();
                    System.out.println("Successfully inserted new product1 with pid " + pid);
                } catch (SQLException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
                }
                break;
            case 2:
                System.out.println("Enter the color of the product 2:");
                String color = scanner.nextLine();
                try {
                    CallableStatement query1 = dbConnection.prepareCall("INSERT INTO PRODUCT2(PID, COLOR) VALUES (?,?)");
                    query1.setInt(1, pid);
                    query1.setString(2, color);
                    query1.executeUpdate();
                    System.out.println("Successfully inserted new product2 with pid " + pid);
                } catch (SQLException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
```

```
                        }
                        break;
                case 3:
                        System.out.println("Enter the weight of the
product 3:");
                        int weight =
Integer.parseInt(scanner.nextLine());
                        try {
                                CallableStatement query1 =
dbConnection.prepareCall("INSERT INTO PRODUCT3(PID, WEIGHT) VALUES
(?,?)");
                                query1.setInt(1, pid);
                                query1.setInt(2, weight);
                                query1.executeUpdate();
                                System.out.println("Successfully
inserted new product3 with pid " + pid);
                        } catch (SQLException e) {
                                // TODO Auto-generated catch block
                                e.printStackTrace();
                        }
                        break;
                }

                break;
           case 3: //Enter a customer associated with some
products (50/day).
                        System.out.println("Enter the customer name:");
                        name = scanner.nextLine();
                        System.out.println("Enter the customer's
address");
                        address = scanner.nextLine();
                        //Execute Query
                        try {
                                CallableStatement query1 =
dbConnection.prepareCall("{CALL INSERT_NEW_CUSTOMER(?,?)}");
                                query1.setString(1, name);
                                query1.setString(2, address);
                                query1.executeUpdate();
                                System.out.println("Successfully inserted
new customer " + name);
                        } catch (SQLException e) {
                                System.out.println("Failed to insert new
customer " + name);
```

```java
                                    e.printStackTrace();
                            }

                            pid = 0;
                            while(pid != -1) {
                                    System.out.println("Enter pid purchased by
" + name +  " or -1 to exit:");
                                    pid = scanner.nextInt();
                                    if (pid != -1) {
                                            try {
                                                    CallableStatement query1 =
dbConnection.prepareCall("{CALL CUSTOMER_PURCHASE(?,?)}");
                                                    query1.setString(1, name);
                                                    query1.setInt(2, pid);
                                                    query1.executeUpdate();
                                                    System.out.println("Successfully
stored " + name + "'s purchase of product " + pid);
                                            } catch (SQLException e) {
                                                    System.out.println("No such
product or customer exists");
                                                    // TODO Auto-generated catch
block
                                                    e.printStackTrace();
                                            }
                                    }
                            }

                            break;
                    case 4: //Create a new account associated with a
product (40/day).
                            //Get input
                            System.out.println("Enter the account number:");
                            int number =
Integer.parseInt(scanner.nextLine());
                            System.out.println("Enter the product id the this
account will track:");
                            pid = Integer.parseInt(scanner.nextLine());
                            System.out.println("Enter the cost of the
product:");
                            int cost = Integer.parseInt(scanner.nextLine());
                            //Execute Query
                            try {
```

```java
                                CallableStatement query1 =
dbConnection.prepareCall("{CALL INSERT_NEW_ACCOUNT(?,?,?,?)}");
                                query1.setInt(1, number);
                                query1.setDate(2, new
Date(System.currentTimeMillis()));
                                query1.setInt(3, cost);
                                query1.setInt(4, pid);
                                query1.executeUpdate();
                                System.out.println("Successfully created
new account " + number);
                        } catch (SQLException e) {
                                System.out.println("Failed to create new
account " + number);
                                e.printStackTrace();
                        }
                        break;

                case 5: //Enter a complaint associated with a customer
and product (30/day).
                        //Get input
                        System.out.println("Enter the complaint id:");
                        int cid = Integer.parseInt(scanner.nextLine());
                        System.out.println("Enter the complaint year
(int):");
                        year = Integer.parseInt(scanner.nextLine());
                        System.out.println("Enter the complaint month
(int):");
                        month = Integer.parseInt(scanner.nextLine());
                        System.out.println("Enter the complaint day
(int):");
                        day = Integer.parseInt(scanner.nextLine());
                        @SuppressWarnings("deprecation")
                        Date date = new Date(year, month, day);
                        System.out.println("Enter the description of the
complaint (max 512 char):");
                        String description = scanner.nextLine();
                        System.out.println("Enter the expected
treatment:");
                        String treatment = scanner.nextLine();
                        System.out.println("Enter the customer's name:");
                        String customerName = scanner.nextLine();
                        System.out.println("Enter the product id:");
                        pid = Integer.parseInt(scanner.nextLine());
```

```java
                        //Execute Query
                        try {
                            CallableStatement query1 =
dbConnection.prepareCall("{CALL INSERT_NEW_COMPLAINT(?,?,?,?,?,?)}");
                            query1.setInt(1, cid);
                            query1.setDate(2, date);
                            query1.setString(3, description);
                            query1.setString(4, treatment);
                            query1.setString(5, customerName);
                            query1.setInt(6, pid);
                            query1.executeUpdate();
                            System.out.println("Successfully created
new complaint " + cid + " with product " + pid);
                        } catch (SQLException e) {
                            System.out.println("Failed to create new
complaint " + cid + " with product " + pid);
                            e.printStackTrace();
                        }
                        break;

                    case 6: //Enter an accident associated with
appropriate employee and product (1/week).
                        //Get input
                        System.out.println("Enter the new accident
number:");
                        number = Integer.parseInt(scanner.nextLine());
                        System.out.println("Enter the year of the
accident:");
                        year = Integer.parseInt(scanner.nextLine());
                        System.out.println("Enter the month of the
accident:");
                        month = Integer.parseInt(scanner.nextLine());
                        System.out.println("Enter the day of the
accident:");
                        day =Integer.parseInt(scanner.nextLine());
                        date = new Date(year, month, day);
                        System.out.println("Enter the number of work days
lost:");
                        int daysLost =
Integer.parseInt(scanner.nextLine());
                        System.out.println("Enter the product id:");
                        pid = Integer.parseInt(scanner.nextLine());
                        System.out.println("Enter the employee's name:");
```

```java
                            name = scanner.nextLine();
                            //Execute Query
                            try {
                                    CallableStatement query1 =
dbConnection.prepareCall("{CALL INSERT_NEW_ACCIDENT(?,?,?,?,?)}");
                                    query1.setInt(1, number);
                                    query1.setDate(2, date);
                                    query1.setInt(3, daysLost);
                                    query1.setInt(4, pid);
                                    query1.setString(5, name);
                                    query1.executeUpdate();
                                    System.out.println("Successfully inserted
new accident " + number);
                            } catch (SQLException e) {
                                    System.out.println("Failed to insert new
accident " + number);
                                    e.printStackTrace();
                            }
                            break;

                    case 7: //Retrieve the date produced and time spent to
produce a particular product (100/day).
                            //Get input
                            System.out.println("Enter the product id:");
                            pid = Integer.parseInt(scanner.nextLine());
                            //Execute Query
                            try {
                                    CallableStatement query1 =
dbConnection.prepareCall("SELECT PRODUCTION_DATE,
PRODUCTION_TIME_MINUTES FROM PRODUCT WHERE PID = ?");
                                    query1.setInt(1, pid);
                                    ResultSet rs = query1.executeQuery();
                                    rs.next();
                                    productionDate =
rs.getDate("PRODUCTION_DATE");
                                    productionTimeMinutes =
rs.getInt("PRODUCTION_TIME_MINUTES");
                                    System.out.println("Production date: " +
productionDate + "\nProduction time (minutes): " +
productionTimeMinutes);
                            } catch (SQLException e) {
                                    // TODO Auto-generated catch block
                                    e.printStackTrace();
```

```java
				}
				break;

		case 8: //Retrieve all products made by a particular
worker (2000/day).
				//Get input
				System.out.println("Enter a workers name:");
				name = scanner.nextLine();
				//Execute Query
				try {
					CallableStatement query1 =
dbConnection.prepareCall("SELECT PID FROM PRODUCT WHERE \"W.NAME\" =
?");
					query1.setString(1, name);
					ResultSet rs = query1.executeQuery();
					System.out.println("Here are the products
made by " + name);

					while (rs.next()) {
						pid = rs.getInt("PID");
						System.out.println(pid);
					}

				} catch (SQLException e) {
					// TODO Auto-generated catch block
					e.printStackTrace();
				}
				break;

		case 9: /*Retrieve the total number of errors a
particular quality controller made. This is the total
				number of products certified by this controller
and got some complaints (400/day).*/
				//Get input
				System.out.println("Enter a quality controller's
name:");
				name = scanner.nextLine();
				//Execute Query
				try {
					CallableStatement query1 =
dbConnection.prepareCall("SELECT COUNT(*) FROM COMPLAINT WHERE
\"PRODUCT.PID\" IN "
						+ "(SELECT \"PID\" FROM PRODUCT
WHERE \"QC.NAME\"=?)");
```

```
                                query1.setString(1, name);
                                ResultSet rs = query1.executeQuery();
                                rs.next();
                                int count = rs.getInt(1);
                                System.out.println(name + " made " + count
+ " errors.");
                        } catch (SQLException e) {
                                // TODO Auto-generated catch block
                                e.printStackTrace();
                        }
                        break;

                case 10: /*Retrieve the total costs of the products in
the product3 category which were repaired at the
                                request of a particular quality
controller (40/day).*/
                        //Get input
                        System.out.println("Enter a quality controller's
name:");
                        name = scanner.nextLine();
                        //Execute Query
                        try {
                                CallableStatement query1 =
dbConnection.prepareCall("SELECT SUM(CASE WHEN NUMBER IN "
                                        + "(SELECT NUMBER FROM ACCOUNT
WHERE 'PRODUCT.PID' IN "
                                        + "(SELECT PID FROM PRODUCT
WHERE 'QC.NAME=?)"
                                        + ") THEN COST ELSE 0 END) FROM
ACCOUNT");
                                query1.setString(1, name);
                                ResultSet rs = query1.executeQuery();
                                rs.next();
                                cost = rs.getInt(1);
                                System.out.println("Total cost: " + cost);
                        } catch (SQLException e) {
                                // TODO Auto-generated catch block
                                e.printStackTrace();
                        }
                        break;

                case 11: /*Retrieve all customers who purchased all
products of a particular color (5/month).*/
```

```java
                        //Get input
                        System.out.println("Enter the color of the
product2:");
                        String color = scanner.nextLine();
                        //Execute Query
                        try {
                                CallableStatement query1 =
dbConnection.prepareCall("SELECT \"CUSTOMER.NAME\" FROM PRODUCT WHERE
"
                                                + "PID IN (SELECT PID FROM
PRODUCT2 WHERE COLOR=?)");
                                query1.setString(1, color);
                                ResultSet rs = query1.executeQuery();
                                System.out.println("Here are the customers
who purchased a " + color + " product2:");
                                while (rs.next()) {
                                        customerName = rs.getString(1);
                                        System.out.println(customerName);
                                }
                        } catch (SQLException e) {
                                // TODO Auto-generated catch block
                                e.printStackTrace();
                        }
                        break;
                case 12: /*Retrieve the total number of work days lost
due to accidents in repairing the products which
                        got complaints (1/month).*/
                        try {
                                CallableStatement query1 =
dbConnection.prepareCall("SELECT SUM(CASE WHEN 'PRODUCT.PID' IN "
                                                + "(SELECT 'PRODUCT.PID' FROM
COMPLAINT) THEN DAYS_LOST ELSE 0 END) FROM ACCIDENT");
                                ResultSet rs = query1.executeQuery();
                                rs.next();
                                int workDaysLost = rs.getInt(1);
                                System.out.println("Work days lost: " +
workDaysLost);
                        } catch (SQLException e) {
                                // TODO Auto-generated catch block
                                e.printStackTrace();
                        }
                        break;
```

```java
            case 13: /*Retrieve all customers who are also workers
(10/month).*/
                        try {
                            CallableStatement query1 =
dbConnection.prepareCall("SELECT \"NAME\" FROM CUSTOMER WHERE NAME IN
"
                                            + "(SELECT \"EMPLOYEE.NAME\"
FROM WORKER)");
                            ResultSet rs = query1.executeQuery();
                            System.out.println("Here are all of the
customers who are also workers:");
                            while (rs.next()) {

System.out.println(rs.getString("name"));
                            }

                        } catch (SQLException e) {
                            // TODO Auto-generated catch block
                            e.printStackTrace();
                        }
                        break;
                case 14: /*Retrieve all the customers who have
purchased the products made or certified or repaired by
                            themselves (5/day).*/
                        try {
                            CallableStatement query1 =
dbConnection.prepareCall("SELECT \"CUSTOMER.NAME\" FROM PRODUCT"
                                            + " WHERE \"CUSTOMER.NAME\" =
\"W.NAME\" "
                                            + " OR \"CUSTOMER.NAME\" =
\"QC.NAME\" "
                                            + " OR \"CUSTOMER.NAME\" =
\"TS.NAME\" ");
                            ResultSet rs = query1.executeQuery();
                            System.out.println("The customers who have
purchased the products made or certified or "
                                            + "repaired by themselves:");
                            while (rs.next()) {
                                System.out.println(rs.getString(1));
                            }
                        } catch (SQLException e) {
                            // TODO Auto-generated catch block
                            e.printStackTrace();
```

```java
                                }
                                break;
                        case 15: /*Retrieve the average cost of all products
made in a particular year*/
                                //Get input
                                System.out.println("Enter a year (yyyy):");
                                year = Integer.parseInt(scanner.nextLine());
                                Date startDate = new Date(year, 1, 1);
                                Date endDate = new Date(year + 1, 1, 1);
                                try {
                                        CallableStatement query1 =
dbConnection.prepareCall("SELECT AVG(COST) FROM \"ACCOUNT\" WHERE
DATE_EST BETWEEN ? AND ?");
                                        query1.setDate(1, startDate);
                                        query1.setDate(2, endDate);
                                        ResultSet rs = query1.executeQuery();
                                        rs.next();
                                        cost = rs.getInt(1);
                                        System.out.println("Average cost of
products made during " + year + ": " + cost);
                                } catch (SQLException e) {
                                        // TODO Auto-generated catch block
                                        e.printStackTrace();
                                }
                                break;
                        case 16: /*Switch the position between a technical
staff and a quality controller (1/ 3 months).*/
                                //Get input
                                System.out.println("Enter a quality controller's
name:");
                                name = scanner.nextLine();
                                try {
                                        CallableStatement query1 =
dbConnection.prepareCall("");
                                        query1.setString(1, name);
                                        ResultSet rs = query1.executeQuery();
                                        rs.next();
                                        cost = rs.getInt(1);
                                        System.out.println("Cost: " + cost);
                                } catch (SQLException e) {
                                        // TODO Auto-generated catch block
                                        e.printStackTrace();
                                }
```

```java
                                break;
                    case 17: /*Delete all accidents whose dates are in
some range (1/day).*/
                            //Get input
                            System.out.println("Enter the starting date year
(yyyy):");
                            year = Integer.parseInt(scanner.nextLine());
                            System.out.println("Enter the starting date month
(mm):");
                            month = Integer.parseInt(scanner.nextLine());
                            System.out.println("Enter the starting date day
(dd):");
                            day = Integer.parseInt(scanner.nextLine());
                            startDate = new Date(year, month, day);

                            System.out.println("Enter the ending date year
(yyyy):");
                            year = Integer.parseInt(scanner.nextLine());
                            System.out.println("Enter the ending date month
(mm):");
                            month = Integer.parseInt(scanner.nextLine());
                            System.out.println("Enter the ending date day
(dd):");
                            day = Integer.parseInt(scanner.nextLine());
                            endDate = new Date(year, month, day);
                            try {
                                CallableStatement query1 =
dbConnection.prepareCall("DELETE FROM ACCIDENT WHERE \"DATE\" BETWEEN
? AND ?");
                                    query1.setDate(1, startDate);
                                    query1.setDate(2, endDate);
                                    query1.executeQuery();
                                    System.out.println("Deleted accidents
between " + startDate.toString() + " and " + endDate.toString());
                            } catch (SQLException e) {
                                    // TODO Auto-generated catch block
                                    e.printStackTrace();
                            }
                            break;
                    case 18: //Import customer information from a file.
                            //Get input
                            System.out.println("Enter the file name
containing the customer information:");
```

```java
                            String fileName = scanner.nextLine();
                            try {
                                    BufferedReader in = new BufferedReader(new
FileReader(fileName));
                                    String line;
                                    while((line = in.readLine()) != null)
                                    {
                                        String[] items = line.split("\\s");
                                        name = items[0];
                                        address = "";
                                        for (int i = 1; i < items.length; i++)
{
                                          address += items[i] + " ";
                                        }
                                        try {
                                                CallableStatement query1 =
dbConnection.prepareCall("{CALL INSERT_NEW_CUSTOMER(?,?)}");
                                                query1.setString(1, name);
                                                query1.setString(2, address);
                                                query1.executeUpdate();
                                                System.out.println("Successfully
inserted new customer " + name);
                                        } catch (SQLException e) {
                                                System.out.println("Failed to
insert new customer " + name);
                                                e.printStackTrace();
                                        }
                                    }
                                    in.close();
                            } catch (IOException e) {
                                    System.out.println("Could not open file " +
fileName);
                                    e.printStackTrace();
                            }
                            break;
                    case 19: //Export customer information to a file.
                            //Get input
                            System.out.println("Enter the file name to output
the customer information to:");
                            fileName = scanner.nextLine();
                            try {
                                    CallableStatement query1 =
dbConnection.prepareCall("SELECT * FROM CUSTOMER");
```

```java
                            ResultSet rs = query1.executeQuery();
                            PrintWriter writer = new
PrintWriter(fileName, "UTF-8");
                            while (rs.next()) {
                                    writer.print(rs.getString(1) + " " +
rs.getString(2) + "\n");
                            }
                            writer.close();
                            System.out.println("Successfully wrote all
customer information to " + fileName);

                    } catch (SQLException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                    } catch (FileNotFoundException e) {
                        System.out.println("Could not open file " +
fileName);
                        e.printStackTrace();
                    } catch (UnsupportedEncodingException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                    }
                    break;
                case 20: //Quit
                    System.out.println("Closing Future Inc DB
Driver...");
                    continu = false;
                    break;
            }
        }
        System.out.println("Closed");
    }

    private static int start(Scanner scanner) {
        String introQueries =
                "Please choose from one of the following
options:\n"
                + "1) Enter a new employee (2/month).\n"
                + "2) Enter a new product associated with the
person who made the product, repaired the product if\n"
                + "it is repaired, or checked the product
(400/day).\n"
```

```
                         + "3) Enter a customer associated with some
products (50/day).\n"
                         + "4) Create a new account associated with a
product (40/day).\n"
                         + "5) Enter a complaint associated with a
customer and product (30/day).\n"
                         + "6) Enter an accident associated with
appropriate employee and product (1/week).\n"
                         + "7) Retrieve the date produced and time spent
to produce a particular product (100/day).\n"
                         + "8) Retrieve all products made by a particular
worker (2000/day).\n"
                         + "9) Retrieve the total number of errors a
particular quality controller made. This is the total\n"
                         + "number of products certified by this
controller and got some complaints (400/day).\n"
                         + "10) Retrieve the total costs of the products
in the product3 category which were repaired at the\n"
                         + "request of a particular quality controller
(40/day).\n"
                         + "11) Retrieve all customers who purchased all
products of a particular color (5/month).\n"
                         + "12) Retrieve the total number of work days
lost due to accidents in repairing the products which\n"
                         + "got complaints (1/month).\n"
                         + "13) Retrieve all customers who are also
workers (10/month).\n"
                         + "14) Retrieve all the customers who have
purchased the products made or certified or repaired by\n"
                         + "themselves (5/day).\n"
                         + "15) Retrieve the average cost of all products
made in a particular year (5/day).\n"
                         + "16) Switch the position between a technical
staff and a quality controller (1/ 3 months).\n"
                         + "17) Delete all accidents whose dates are in
some range (1/day).\n"
                         + "18) Import customer information from a
file.\n"
                         + "19) Export customer information to a file.\n"
                         + "20) Quit\n";
            System.out.println(introQueries);
            return Integer.parseInt(scanner.nextLine());
        }
```

}

6.1 Query 1 (x10)

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if it is repaired, or checked the product (400/day).
3) Enter a customer associated with some products (50/day).
4) Create a new account associated with a product (40/day).
5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).
8) Retrieve all products made by a particular worker (2000/day).
9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).
10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).
11) Retrieve all customers who purchased all products of a particular color (5/month).
12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).
13) Retrieve all customers who are also workers (10/month).
14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).
15) Retrieve the average cost of all products made in a particular year (5/day).
16) Switch the position between a technical staff and a quality controller (1/ 3 months).
17) Delete all accidents whose dates are in some range (1/day).
18) Import customer information from a file.
19) Export customer information to a file.
20) Quit
1
Enter the new employee's name:
Luigi
Enter the new employee's address:
232 E Market St.
Enter the type of employee (Worker, Technical Staff, or Quality Controller:
Worker
Enter the production limit for Luigi:
10
Successfully inserted new worker Luigi.
Please choose from one of the following options:
display start menu
1
Enter the new employee's name:
Bowser

Enter the new employee's address:

800 S Devils Way

Enter the type of employee (Worker, Technical Staff, or Quality Controller:

Technical Staff

Enter Bowser's highest level of education (BS, MS, or PhD):

BS

Enter Bowser's technical position:

Manager

Successfully inserted new technical staff Bowser.

Please choose from one of the following options:

display start menu

1

Enter the new employee's name:

Peach

Enter the new employee's address:

399 W Heavenly Ct.

Enter the type of employee (Worker, Technical Staff, or Quality Controller:

Quality Controller

Enter the product type for Peach (1, 2, or 3):

1

Successfully inserted new quality controller Peach.

Please choose from one of the following options:

display start menu

1

Enter the new employee's name:

Riley

Enter the new employee's address:

900 E Lindsey St

Enter the type of employee (Worker, Technical Staff, or Quality Controller:

Worker

Enter the production limit for Riley:

11

Successfully inserted new worker Riley.

Please choose from one of the following options:

display start menu

1

Enter the new employee's name:

Mario

Enter the new employee's address:

232 E Market St.

Enter the type of employee (Worker, Technical Staff, or Quality Controller:

Technical Staff

Enter Mario's highest level of education (BS, MS, or PhD):

MS

Enter Mario's technical position:

Director

Successfully inserted new technical staff Mario.

Please choose from one of the following options:

display start menu

1

Enter the new employee's name:

Wario

Enter the new employee's address:

648 E Stockholm Lane

Enter the type of employee (Worker, Technical Staff, or Quality Controller:

Quality Controller

Enter the product type for Wario (1, 2, or 3):

2

Successfully inserted new quality controller Wario.

Please choose from one of the following options:

display start menu

1

Enter the new employee's name:

Waluigi

Enter the new employee's address:

606 Sundance Dr.

Enter the type of employee (Worker, Technical Staff, or Quality Controller:

Worker

Enter the production limit for Waluigi:

12

Successfully inserted new worker Waluigi.

Please choose from one of the following options:

display start menu

1

Enter the new employee's name:

Lakitu

Enter the new employee's address:

435 Cloudy Way

Enter the type of employee (Worker, Technical Staff, or Quality Controller:

Technical Staff

Enter Lakitu's highest level of education (BS, MS, or PhD):

PhD

Enter Lakitu's technical position:

Associate

Successfully inserted new technical staff Lakitu.

Please choose from one of the following options:

display start menu
1
Enter the new employee's name:
Daisy
Enter the new employee's address:
989 Flowery Ave
Enter the type of employee (Worker, Technical Staff, or Quality Controller:
Quality Controller
Enter the product type for Daisy (1, 2, or 3):
3
Successfully inserted new quality controller Daisy.
Please choose from one of the following options:
display start menu
1
Enter the new employee's name:
Yoshi
Enter the new employee's address:
765 Hopping Hill
Enter the type of employee (Worker, Technical Staff, or Quality Controller:
Worker
Enter the production limit for Yoshi:
14
Successfully inserted new worker Yoshi.

6.2 Query 2 (x10)

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if it is repaired, or checked the product (400/day).
3) Enter a customer associated with some products (50/day).
4) Create a new account associated with a product (40/day).
5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).
8) Retrieve all products made by a particular worker (2000/day).
9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).
10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).
11) Retrieve all customers who purchased all products of a particular color (5/month).
12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).

13) Retrieve all customers who are also workers (10/month).

14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).

15) Retrieve the average cost of all products made in a particular year (5/day).

16) Switch the position between a technical staff and a quality controller (1/ 3 months).

17) Delete all accidents whose dates are in some range (1/day).

18) Import customer information from a file.

19) Export customer information to a file.

20) Quit

2

Enter a pid:

1

Enter the production year (int):

2000

Enter the production month (int):

2

Enter the production day (int):

1

Enter a production time (minutes):

6

Enter a size:

s

Enter a worker name:

Waluigi

Enter a quality controller name:

Peach

Enter a technical staff name:

Bowser

Enter the repair year (int):

2001

Enter the repair month (int):

2

Enter the repair day (int):

1

Enter the product type (1, 2, or 3)

1

Enter the name of the software name for product 1:

Adobe

Successfully inserted new product1 with pid 1

Please choose from one of the following options:

*** display options 1-20 again ***

2

Enter a pid:

2

Enter the production year (int):

2000

Enter the production month (int):

2

Enter the production day (int):

2

Enter a production time (minutes):

9

Enter a size:

m

Enter a worker name:

Luigi

Enter a quality controller name:

Wario

Enter a technical staff name:

Mario

Enter the repair year (int):

2001

Enter the repair month (int):

2

Enter the repair day (int):

2

Enter the product type (1, 2, or 3)

2

Enter the color of the product 2:

Red

Successfully inserted new product2 with pid 2

Please choose from one of the following options:

*** display options 1-20 again ***

2

Enter a pid:

3

Enter the production year (int):

2000

Enter the production month (int):

2

Enter the production day (int):

3

Enter a production time (minutes):

4

Enter a size:

l

Enter a worker name:

Riley

Enter a quality controller name:

Daisy

Enter a technical staff name:

Lakitu

Enter the repair year (int):

2001

Enter the repair month (int):

2

Enter the repair day (int):

3

Enter the product type (1, 2, or 3):

3

Enter the weight of the product 3:

4

Successfully inserted new product3 with pid 3

Please choose from one of the following options:

*** display options 1-20 again ***

2

Enter a pid:

4

Enter the production year (int):

2000

Enter the production month (int):

2

Enter the production day (int):

4

Enter a production time (minutes):

9

Enter a size:

s

Enter a worker name:

Yoshi

Enter a quality controller name:

Peach

Enter a technical staff name:

Bowser

Enter the repair year (int):

2001

Enter the repair month (int):

2

Enter the repair day (int):

4
Enter the product type (1, 2, or 3)
1
Enter the name of the software name for product 1:
Microsoft
Successfully inserted new product1 with pid 4
Please choose from one of the following options:
*** display options 1-20 again ***
2
Enter a pid:
5
Enter the production year (int):
2000
Enter the production month (int):
2
Enter the production day (int):
5
Enter a production time (minutes):
9
Enter a size:
m
Enter a worker name:
Waluigi
Enter a quality controller name:
Wario
Enter a technical staff name:
Mario
Enter the repair year (int):
2001
Enter the repair month (int):
2
Enter the repair day (int):
5
Enter the product type (1, 2, or 3)
2
Enter the color of the product 2:
Orange
Successfully inserted new product2 with pid 5
Please choose from one of the following options:
*** display options 1-20 again ***
2
Enter a pid:
6

Enter the production year (int):

2000

Enter the production month (int):

2

Enter the production day (int):

6

Enter a production time (minutes):

11

Enter a size:

l

Enter a worker name:

Luigi

Enter a quality controller name:

Daisy

Enter a technical staff name:

Lakitu

Enter the repair year (int):

2001

Enter the repair month (int):

2

Enter the repair day (int):

6

Enter the product type (1, 2, or 3)

3

Enter the weight of the product 3:

7

Successfully inserted new product3 with pid 6

Please choose from one of the following options:

*** display options 1-20 again ***

2

Enter a pid:

7

Enter the production year (int):

2000

Enter the production month (int):

2

Enter the production day (int):

7

Enter a production time (minutes):

7

Enter a size:

s

Enter a worker name:

Riley
Enter a quality controller name:
Peach
Enter a technical staff name:
Bowser
Enter the repair year (int):
2001
Enter the repair month (int):
2
Enter the repair day (int):
7
Enter the product type (1, 2, or 3)
1
Enter the name of the software name for product 1:
Apple
Successfully inserted new product1 with pid 7
Please choose from one of the following options:
*** display options 1-20 again ***
2
Enter a pid:
8
Enter the production year (int):
2000
Enter the production month (int):
2
Enter the production day (int):
8
Enter a production time (minutes):
12
Enter a size:
m
Enter a worker name:
Yoshi
Enter a quality controller name:
Wario
Enter a technical staff name:
Mario
Enter the repair year (int):
2001
Enter the repair month (int):
2
Enter the repair day (int):
8

Enter the product type (1, 2, or 3)

2

Enter the color of the product 2:

Yellow

Successfully inserted new product2 with pid 8

Please choose from one of the following options:

*** display options 1-20 again ***

2

Enter a pid:

9

Enter the production year (int):

2000

Enter the production month (int):

2

Enter the production day (int):

9

Enter a production time (minutes):

14

Enter a size:

l

Enter a worker name:

Waluigi

Enter a quality controller name:

Daisy

Enter a technical staff name:

Lakitu

Enter the repair year (int):

2001

Enter the repair month (int):

2

Enter the repair day (int):

9

Enter the product type (1, 2, or 3)

3

Enter the weight of the product 3:

9

Successfully inserted new product3 with pid 9

Please choose from one of the following options:

*** display options 1-20 again ***

2

Enter a pid:

10

Enter the production year (int):

2000
Enter the production month (int):
2
Enter the production day (int):
10
Enter a production time (minutes):
8
Enter a size:
s
Enter a worker name:
Luigi
Enter a quality controller name:
Peach
Enter a technical staff name:
Bowser
Enter the repair year (int):
2001
Enter the repair month (int):
2
Enter the repair day (int):
10
Enter the product type (1, 2, or 3)
1
Enter the name of the software name for product 1:
Android
Successfully inserted new product1 with pid 10
Please choose from one of the following options:
*** display options 1-20 again ***

6.3 Query 3 (x10)

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if it is repaired, or checked the product (400/day).
3) Enter a customer associated with some products (50/day).
4) Create a new account associated with a product (40/day).
5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).
8) Retrieve all products made by a particular worker (2000/day).
9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).

46

10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).

11) Retrieve all customers who purchased all products of a particular color (5/month).

12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).

13) Retrieve all customers who are also workers (10/month).

14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).

15) Retrieve the average cost of all products made in a particular year (5/day).

16) Switch the position between a technical staff and a quality controller (1/ 3 months).

17) Delete all accidents whose dates are in some range (1/day).

18) Import customer information from a file.

19) Export customer information to a file.

20) Quit

3

Enter the customer name:

Riley

Enter the customer's address

900 E Lindsey St.

Successfully inserted new customer Riley

Enter pid purchased by Riley or -1 to exit:

1

Successfully stored Riley's purchase of product 1

Enter pid purchased by Riley or -1 to exit:

2

Successfully stored Riley's purchase of product 2

Enter pid purchased by Riley or -1 to exit:

3

Successfully stored Riley's purchase of product 3

Enter pid purchased by Riley or -1 to exit:

-1

Please choose from one of the following options:

*** display options 1-20 again ***

3

Enter the customer name:

Yoshi

Enter the customer's address

232 E Market St.

Successfully inserted new customer Yoshi

Enter pid purchased by Yoshi or -1 to exit:

-1

Please choose from one of the following options:

*** display options 1-20 again ***

3
Enter the customer name:
Bowser
Enter the customer's address
800 S Devils Way
Successfully inserted new customer Bowser
Enter pid purchased by Bowser or -1 to exit:
-1
Please choose from one of the following options:
*** display options 1-20 again ***
3
Enter the customer name:
Peach
Enter the customer's address
399 W Heavenly Ct.
Successfully inserted new customer Peach
Enter pid purchased by Peach or -1 to exit:
-1
Please choose from one of the following options:
*** display options 1-20 again ***
3
Enter the customer name:
Luigi
Enter the customer's address
232 E Market St.
Successfully inserted new customer Luigi
Enter pid purchased by Luigi or -1 to exit:
4
Successfully stored Luigi's purchase of product 4
Enter pid purchased by Luigi or -1 to exit:
-1
Please choose from one of the following options:
*** display options 1-20 again ***
3
Enter the customer name:
Mario
Enter the customer's address
232 E Market St.
Successfully inserted new customer Mario
Enter pid purchased by Mario or -1 to exit:
5
Successfully stored Mario's purchase of product 5
Enter pid purchased by Mario or -1 to exit:

-1

Please choose from one of the following options:

*** display options 1-20 again ***

3

Enter the customer name:

Jim

Enter the customer's address

394 Mulberry St

Successfully inserted new customer Jim

Enter pid purchased by Jim or -1 to exit:

6

Successfully stored Jim's purchase of product 6

Enter pid purchased by Jim or -1 to exit:

-1

Please choose from one of the following options:

*** display options 1-20 again ***

3

Enter the customer name:

John

Enter the customer's address

432 Hacker Way

Successfully inserted new customer John

Enter pid purchased by John or -1 to exit:

7

Successfully stored John's purchase of product 7

Enter pid purchased by John or -1 to exit:

-1

Please choose from one of the following options:

*** display options 1-20 again ***

3

Enter the customer name:

Bob

Enter the customer's address

3025 Balmoral Dr.

Successfully inserted new customer Bob

Enter pid purchased by Bob or -1 to exit:

8

Successfully stored Bob's purchase of product 8

Enter pid purchased by Bob or -1 to exit:

-1

Please choose from one of the following options:

*** display options 1-20 again ***

3

Enter the customer name:
Jeremy
Enter the customer's address
525 Realty St.
Successfully inserted new customer Jeremy
Enter pid purchased by Jeremy or -1 to exit:
9
Successfully stored Jeremy's purchase of product 9
Enter pid purchased by Jeremy or -1 to exit:
-1
Please choose from one of the following options:
*** display options 1-20 again ***
3
Enter the customer name:
Ryan
Enter the customer's address
123 Skinny St.
Successfully inserted new customer Ryan
Enter pid purchased by Ryan or -1 to exit:
10
Successfully stored Ryan's purchase of product 10
Enter pid purchased by Ryan or -1 to exit:
-1
Please choose from one of the following options:
*** display options 1-20 again ***

6.4 Query 4 (x10)

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if
it is repaired, or checked the product (400/day).
3) Enter a customer associated with some products (50/day).
4) Create a new account associated with a product (40/day).
5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).
8) Retrieve all products made by a particular worker (2000/day).
9) Retrieve the total number of errors a particular quality controller made. This is the total
number of products certified by this controller and got some complaints (400/day).
10) Retrieve the total costs of the products in the product3 category which were repaired at the
request of a particular quality controller (40/day).
11) Retrieve all customers who purchased all products of a particular color (5/month).

12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).

13) Retrieve all customers who are also workers (10/month).

14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).

15) Retrieve the average cost of all products made in a particular year (5/day).

16) Switch the position between a technical staff and a quality controller (1/ 3 months).

17) Delete all accidents whose dates are in some range (1/day).

18) Import customer information from a file.

19) Export customer information to a file.

20) Quit

4

Enter the account number:

1

Enter the product id the this account will track:

1

Enter the cost of the product:

1

Successfully created new account 1

Please choose from one of the following options:

*** display options 1-20 again ***

4

Enter the account number:

2

Enter the product id the this account will track:

2

Enter the cost of the product:

2

Successfully created new account 2

Please choose from one of the following options:

*** display options 1-20 again ***

4

Enter the account number:

3

Enter the product id the this account will track:

3

Enter the cost of the product:

3

Successfully created new account 3

Please choose from one of the following options:

*** display options 1-20 again ***

4

Enter the account number:

4

Enter the product id the this account will track:

4

Enter the cost of the product:

4

Successfully created new account 4

Please choose from one of the following options:

*** display options 1-20 again ***

4

Enter the account number:

5

Enter the product id the this account will track:

5

Enter the cost of the product:

5

Successfully created new account 5

Please choose from one of the following options:

*** display options 1-20 again ***

4

Enter the account number:

6

Enter the product id the this account will track:

6

Enter the cost of the product:

6

Successfully created new account 6

Please choose from one of the following options:

*** display options 1-20 again ***

4

Enter the account number:

7

Enter the product id the this account will track:

7

Enter the cost of the product:

7

Successfully created new account 7

Please choose from one of the following options:

*** display options 1-20 again ***

4

Enter the account number:

8

Enter the product id the this account will track:

8

Enter the cost of the product:

8

Successfully created new account 8

Please choose from one of the following options:

*** display options 1-20 again ***

4

Enter the account number:

9

Enter the product id the this account will track:

9

Enter the cost of the product:

9

Successfully created new account 9

Please choose from one of the following options:

*** display options 1-20 again ***

4

Enter the account number:

10

Enter the product id the this account will track:

10

Enter the cost of the product:

10

Successfully created new account 10

6.5 Query 5 (x3)

Please choose from one of the following options:

1) Enter a new employee (2/month).

2) Enter a new product associated with the person who made the product, repaired the product if
it is repaired, or checked the product (400/day).

3) Enter a customer associated with some products (50/day).

4) Create a new account associated with a product (40/day).

5) Enter a complaint associated with a customer and product (30/day).

6) Enter an accident associated with appropriate employee and product (1/week).

7) Retrieve the date produced and time spent to produce a particular product (100/day).

8) Retrieve all products made by a particular worker (2000/day).

9) Retrieve the total number of errors a particular quality controller made. This is the total
number of products certified by this controller and got some complaints (400/day).

10) Retrieve the total costs of the products in the product3 category which were repaired at the
request of a particular quality controller (40/day).

11) Retrieve all customers who purchased all products of a particular color (5/month).

12) Retrieve the total number of work days lost due to accidents in repairing the products which
got complaints (1/month).

13) Retrieve all customers who are also workers (10/month).

14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).

15) Retrieve the average cost of all products made in a particular year (5/day).

16) Switch the position between a technical staff and a quality controller (1/ 3 months).

17) Delete all accidents whose dates are in some range (1/day).

18) Import customer information from a file.

19) Export customer information to a file.

20) Quit

5

Enter the complaint id:

1

Enter the complaint year (int):

2001

Enter the complaint month (int):

2

Enter the complaint day (int):

1

Enter the description of the complaint (max 512 char):

There was a broken arm on the product in the box

Enter the expected treatment:

exchange

Enter the customer's name:

Riley

Enter the product id:

1

Successfully created new complaint 1 with product 1

Please choose from one of the following options:

*** display options 1-20 again ***

5

Enter the complaint id:

2

Enter the complaint year (int):

2001

Enter the complaint month (int):

2

Enter the complaint day (int):

2

Enter the description of the complaint (max 512 char):

Scratches on product

Enter the expected treatment:
reimbursement
Enter the customer's name:
Bob
Enter the product id:
8
Successfully created new complaint 2 with product 8

Please choose from one of the following options:
*** display options 1-20 again ***
5
Enter the complaint id:
3
Enter the complaint year (int):
2001
Enter the complaint month (int):
2
Enter the complaint day (int):
3
Enter the description of the complaint (max 512 char):
crack in product
Enter the expected treatment:
exchange
Enter the customer's name:
Ryan
Enter the product id:
10
Successfully created new complaint 3 with product 10

6.6 Query 6 (x3)

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if
it is repaired, or checked the product (400/day).
3) Enter a customer associated with some products (50/day).
4) Create a new account associated with a product (40/day).
5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).

8) Retrieve all products made by a particular worker (2000/day).

9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).

10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).

11) Retrieve all customers who purchased all products of a particular color (5/month).

12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).

13) Retrieve all customers who are also workers (10/month).

14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).

15) Retrieve the average cost of all products made in a particular year (5/day).

16) Switch the position between a technical staff and a quality controller (1/ 3 months).

17) Delete all accidents whose dates are in some range (1/day).

18) Import customer information from a file.

19) Export customer information to a file.

20) Quit

6

Enter the new accident number:

1

Enter the year of the accident:

2001

Enter the month of the accident:

2

Enter the day of the accident:

9

Enter the number of work days lost:

3

Enter the product id:

1

Enter the employee's name:

Riley

Successfully inserted new accident 1

Please choose from one of the following options:

*** display options 1-20 again ***

6

Enter the new accident number:

2

Enter the year of the accident:

2001

Enter the month of the accident:

2
Enter the day of the accident:
10
Enter the number of work days lost:
6
Enter the product id:
2
Enter the employee's name:
Mario
Successfully inserted new accident 2

Please choose from one of the following options:
*** display options 1-20 again ***
6
Enter the new accident number:
3
Enter the year of the accident:
2001
Enter the month of the accident:
2
Enter the day of the accident:
7
Enter the number of work days lost:
1
Enter the product id:
3
Enter the employee's name:
Lakitu
Successfully inserted new accident 3

6.7 Query 7

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if
it is repaired, or checked the product (400/day).
3) Enter a customer associated with some products (50/day).
4) Create a new account associated with a product (40/day).
5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).

8) Retrieve all products made by a particular worker (2000/day).

9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).

10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).

11) Retrieve all customers who purchased all products of a particular color (5/month).

12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).

13) Retrieve all customers who are also workers (10/month).

14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).

15) Retrieve the average cost of all products made in a particular year (5/day).

16) Switch the position between a technical staff and a quality controller (1/ 3 months).

17) Delete all accidents whose dates are in some range (1/day).

18) Import customer information from a file.

19) Export customer information to a file.

20) Quit

7

Enter the product id:

1

Production date: 2000-03-01

Production time (minutes): 6

Please choose from one of the following options:

*** display options 1-20 again ***

7

Enter the product id:

2

Production date: 2000-03-02

Production time (minutes): 9

Please choose from one of the following options:

*** display options 1-20 again ***

7

Enter the product id:

3

Production date: 2000-03-03

Production time (minutes): 4

6.8 Query 8

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if it is repaired, or checked the product (400/day).
3) Enter a customer associated with some products (50/day).
4) Create a new account associated with a product (40/day).
5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).
8) Retrieve all products made by a particular worker (2000/day).
9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).
10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).
11) Retrieve all customers who purchased all products of a particular color (5/month).
12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).
13) Retrieve all customers who are also workers (10/month).
14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).
15) Retrieve the average cost of all products made in a particular year (5/day).
16) Switch the position between a technical staff and a quality controller (1/ 3 months).
17) Delete all accidents whose dates are in some range (1/day).
18) Import customer information from a file.
19) Export customer information to a file.
20) Quit
8
Enter a workers name:
Riley
Here are the products made by Riley
3
7

Please choose from one of the following options:
*** display options 1-20 again ***
8
Enter a workers name:
Waluigi
Here are the products made by Waluigi
1
5
9

Please choose from one of the following options:
*** display options 1-20 again ***
8
Enter a workers name:
Yoshi
Here are the products made by Yoshi
4
8

6.9 Query 9

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if it is repaired, or checked the product (400/day).
3) Enter a customer associated with some products (50/day).
4) Create a new account associated with a product (40/day).
5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).
8) Retrieve all products made by a particular worker (2000/day).
9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).
10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).
11) Retrieve all customers who purchased all products of a particular color (5/month).
12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).
13) Retrieve all customers who are also workers (10/month).
14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).
15) Retrieve the average cost of all products made in a particular year (5/day).
16) Switch the position between a technical staff and a quality controller (1/ 3 months).
17) Delete all accidents whose dates are in some range (1/day).
18) Import customer information from a file.
19) Export customer information to a file.
20) Quit
9
Enter a quality controller's name:
Peach

Peach made 2 errors.

Please choose from one of the following options:
*** display options 1-20 again ***
9
Enter a quality controller's name:
Wario
Wario made 1 errors.

Please choose from one of the following options:
*** display options 1-20 again ***
9
Enter a quality controller's name:
Daisy
Daisy made 0 errors.

6.10 Query 10

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if it is repaired, or checked the product (400/day).
3) Enter a customer associated with some products (50/day).
4) Create a new account associated with a product (40/day).
5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).
8) Retrieve all products made by a particular worker (2000/day).
9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).
10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).
11) Retrieve all customers who purchased all products of a particular color (5/month).
12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).
13) Retrieve all customers who are also workers (10/month).
14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).
15) Retrieve the average cost of all products made in a particular year (5/day).
16) Switch the position between a technical staff and a quality controller (1/ 3 months).

17) Delete all accidents whose dates are in some range (1/day).
18) Import customer information from a file.
19) Export customer information to a file.
20) Quit
10
Enter a quality controller's name:
Daisy
Total cost: 18

*** display options 1-20 again ***
10
Enter a quality controller's name:
Wario
Total cost: 0

*** display options 1-20 again ***
10
Enter a quality controller's name:
Peach
Total cost: 0

6.11 Query 11 (x3)

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if it is repaired, or checked the product (400/day).
3) Enter a customer associated with some products (50/day).
4) Create a new account associated with a product (40/day).
5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).
8) Retrieve all products made by a particular worker (2000/day).
9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).
10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).
11) Retrieve all customers who purchased all products of a particular color (5/month).
12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).

13) Retrieve all customers who are also workers (10/month).

14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).

15) Retrieve the average cost of all products made in a particular year (5/day).

16) Switch the position between a technical staff and a quality controller (1/ 3 months).

17) Delete all accidents whose dates are in some range (1/day).

18) Import customer information from a file.

19) Export customer information to a file.

20) Quit

11

Enter the color of the product2:

Red

Here are the customers who purchased a Red product2:

Riley

Please choose from one of the following options:

*** display options 1-20 again ***

11

Enter the color of the product2:

Orange

Here are the customers who purchased a Orange product2:

Mario

Please choose from one of the following options:

*** display options 1-20 again ***

11

Enter the color of the product2:

Yellow

Here are the customers who purchased a Yellow product2:

Bob

6.12 Query 12

Please choose from one of the following options:

1) Enter a new employee (2/month).

2) Enter a new product associated with the person who made the product, repaired the product if it is repaired, or checked the product (400/day).

3) Enter a customer associated with some products (50/day).

4) Create a new account associated with a product (40/day).

5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).
8) Retrieve all products made by a particular worker (2000/day).
9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).
10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).
11) Retrieve all customers who purchased all products of a particular color (5/month).
12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).
13) Retrieve all customers who are also workers (10/month).
14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).
15) Retrieve the average cost of all products made in a particular year (5/day).
16) Switch the position between a technical staff and a quality controller (1/ 3 months).
17) Delete all accidents whose dates are in some range (1/day).
18) Import customer information from a file.
19) Export customer information to a file.
20) Quit
12
Work days lost: 10

6.13 Query 13

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if it is repaired, or checked the product (400/day).
3) Enter a customer associated with some products (50/day).
4) Create a new account associated with a product (40/day).
5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).
8) Retrieve all products made by a particular worker (2000/day).
9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).
10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).
11) Retrieve all customers who purchased all products of a particular color (5/month).
12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).
13) Retrieve all customers who are also workers (10/month).

14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).

15) Retrieve the average cost of all products made in a particular year (5/day).

16) Switch the position between a technical staff and a quality controller (1/ 3 months).

17) Delete all accidents whose dates are in some range (1/day).

18) Import customer information from a file.

19) Export customer information to a file.

20) Quit

13

Here are all of the customers who are also workers:

Luigi

Riley

Yoshi

6.14 Query 14

Please choose from one of the following options:

1) Enter a new employee (2/month).

2) Enter a new product associated with the person who made the product, repaired the product if it is repaired, or checked the product (400/day).

3) Enter a customer associated with some products (50/day).

4) Create a new account associated with a product (40/day).

5) Enter a complaint associated with a customer and product (30/day).

6) Enter an accident associated with appropriate employee and product (1/week).

7) Retrieve the date produced and time spent to produce a particular product (100/day).

8) Retrieve all products made by a particular worker (2000/day).

9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).

10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).

11) Retrieve all customers who purchased all products of a particular color (5/month).

12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).

13) Retrieve all customers who are also workers (10/month).

14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).

15) Retrieve the average cost of all products made in a particular year (5/day).

16) Switch the position between a technical staff and a quality controller (1/ 3 months).

17) Delete all accidents whose dates are in some range (1/day).

18) Import customer information from a file.

19) Export customer information to a file.

20) Quit

14

The customers who have purchased the products made or certified or repaired by themselves:
Riley
Mario

6.15 Query 15

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if it is repaired, or checked the product (400/day).
3) Enter a customer associated with some products (50/day).
4) Create a new account associated with a product (40/day).
5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).
8) Retrieve all products made by a particular worker (2000/day).
9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).
10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).
11) Retrieve all customers who purchased all products of a particular color (5/month).
12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).
13) Retrieve all customers who are also workers (10/month).
14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).
15) Retrieve the average cost of all products made in a particular year (5/day).
16) Switch the position between a technical staff and a quality controller (1/ 3 months).
17) Delete all accidents whose dates are in some range (1/day).
18) Import customer information from a file.
19) Export customer information to a file.
20) Quit
15
Enter a year (yyyy):
2001
Average cost of products during 2001: 6

6.17 Query 17

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if it is repaired, or checked the product (400/day).

3) Enter a customer associated with some products (50/day).

4) Create a new account associated with a product (40/day).

5) Enter a complaint associated with a customer and product (30/day).

6) Enter an accident associated with appropriate employee and product (1/week).

7) Retrieve the date produced and time spent to produce a particular product (100/day).

8) Retrieve all products made by a particular worker (2000/day).

9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).

10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).

11) Retrieve all customers who purchased all products of a particular color (5/month).

12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).

13) Retrieve all customers who are also workers (10/month).

14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).

15) Retrieve the average cost of all products made in a particular year (5/day).

16) Switch the position between a technical staff and a quality controller (1/ 3 months).

17) Delete all accidents whose dates are in some range (1/day).

18) Import customer information from a file.

19) Export customer information to a file.

20) Quit

17

Enter the starting date year (yyyy):

2001

Enter the starting date month (mm):

2

Enter the starting date day (dd):

1

Enter the ending date year (yyyy):

2001

Enter the ending date month (mm):

2

Enter the ending date day (dd):

9

Deleted accidents between 2001-02-01 and 2001-2-9

6.18 Query 18

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if it is repaired, or checked the product (400/day).
3) Enter a customer associated with some products (50/day).
4) Create a new account associated with a product (40/day).
5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).
8) Retrieve all products made by a particular worker (2000/day).
9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).
10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).
11) Retrieve all customers who purchased all products of a particular color (5/month).
12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).
13) Retrieve all customers who are also workers (10/month).
14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).
15) Retrieve the average cost of all products made in a particular year (5/day).
16) Switch the position between a technical staff and a quality controller (1/ 3 months).
17) Delete all accidents whose dates are in some range (1/day).
18) Import customer information from a file.
19) Export customer information to a file.
20) Quit
18
Enter the file name containing the customer information:
/Users/Riley/workspace/FutureDB/src/input.txt
Successfully inserted new customer Bowser
Successfully inserted new customer Riley
Successfully inserted new customer Yoshi
Successfully inserted new customer Peach
Successfully inserted new customer Luigi
Successfully inserted new customer Mario
Successfully inserted new customer Jim
Successfully inserted new customer John
Successfully inserted new customer Bob
Successfully inserted new customer Jeremy
Successfully inserted new customer Ryan

6.19 Query 19

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if it is repaired, or checked the product (400/day).
3) Enter a customer associated with some products (50/day).
4) Create a new account associated with a product (40/day).
5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).
8) Retrieve all products made by a particular worker (2000/day).
9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).
10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).
11) Retrieve all customers who purchased all products of a particular color (5/month).
12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).
13) Retrieve all customers who are also workers (10/month).
14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).
15) Retrieve the average cost of all products made in a particular year (5/day).
16) Switch the position between a technical staff and a quality controller (1/ 3 months).
17) Delete all accidents whose dates are in some range (1/day).
18) Import customer information from a file.
19) Export customer information to a file.
20) Quit
19
Enter the file name to output the customer information to:
/Users/Riley/workspace/FutureDB/src/output.txt
Successfully wrote all customer information to /Users/Riley/workspace/FutureDB/src/output.txt

6.20 Query 20

Please choose from one of the following options:
1) Enter a new employee (2/month).
2) Enter a new product associated with the person who made the product, repaired the product if it is repaired, or checked the product (400/day).
3) Enter a customer associated with some products (50/day).
4) Create a new account associated with a product (40/day).
5) Enter a complaint associated with a customer and product (30/day).
6) Enter an accident associated with appropriate employee and product (1/week).
7) Retrieve the date produced and time spent to produce a particular product (100/day).
8) Retrieve all products made by a particular worker (2000/day).

9) Retrieve the total number of errors a particular quality controller made. This is the total number of products certified by this controller and got some complaints (400/day).

10) Retrieve the total costs of the products in the product3 category which were repaired at the request of a particular quality controller (40/day).

11) Retrieve all customers who purchased all products of a particular color (5/month).

12) Retrieve the total number of work days lost due to accidents in repairing the products which got complaints (1/month).

13) Retrieve all customers who are also workers (10/month).

14) Retrieve all the customers who have purchased the products made or certified or repaired by themselves (5/day).

15) Retrieve the average cost of all products made in a particular year (5/day).

16) Switch the position between a technical staff and a quality controller (1/ 3 months).

17) Delete all accidents whose dates are in some range (1/day).

18) Import customer information from a file.

19) Export customer information to a file.

20) Quit

20

Closing Future Inc DB Driver...

Closed