

Тема 5

Проект по база от данни за Сервиз за битова техника

Изготвил: Мария Маргаритова, фн: 1MI0700004

1. Обхват на модела. Дефиниране на задачата.

(Сервиз за битова техника) Информационна система за сервиз за битова техника съхранява информация за техники и ремонтите, извършени от тях. За всеки ремонтиран от сервиза уред се съхранява задължително следната информация: уникален идентификатор, категория (напр. пералня, телевизор и т.н.), производител (до 50 символа), модел (до 20 символа), име на клиент, а ако е известна и годината на производство (цяло число, по-голямо от 1900). За всеки техник се съхранява задължително следната информация: уникален идентификатор, име (до 100 символа), ЕГН, както и категории уреди, които може да ремонтира. Един техник може да извършва много ремонти на уреди. Един уред може да бъде поправян много пъти от различен техник. За всяка поправка се пази дата и цена. За дадена категория уреди може да има няколко техники в сервиза.

2. Множества от същности и техните атрибути

Ремонтиран уред: уникален идентификатор, категория, производител, модел, име на клиент, година на производство.

Техник: уникален идентификатор, име, ЕГН, категории уреди.

3. Домейн на атрибутите

Ремонтиран уред: уникален идентификатор: низ, категория: низ, производител: низ (до 50 символа), модел: низ (до 20 символа), име на клиент: низ, година на производство: цяло положително число, по-голямо от 1900.

Техник: уникален идентификатор: низ, име: низ (до 100 символа), ЕГН: низ, категории уреди: низ.

4. Връзки

Връзката между техник уред е много към много, защото, един техник може да извършва много ремонти на уреди и един уред може да бъде поправян много пъти от различен техник. За дадена категория уреди може да има няколко техники в сервиза.

5. Ограничения по единствена стойност, референтна цялостност и друг тип ограничения

Ремонтиран уред: уникален идентификатор(id) – еднозначно определя уреда

Техник: уникален идентификатор(id) – еднозначно определя техника

6. Правила и проверки

За уред - Производител (до 50 символа)

За уред - Модел (до 20 символа)

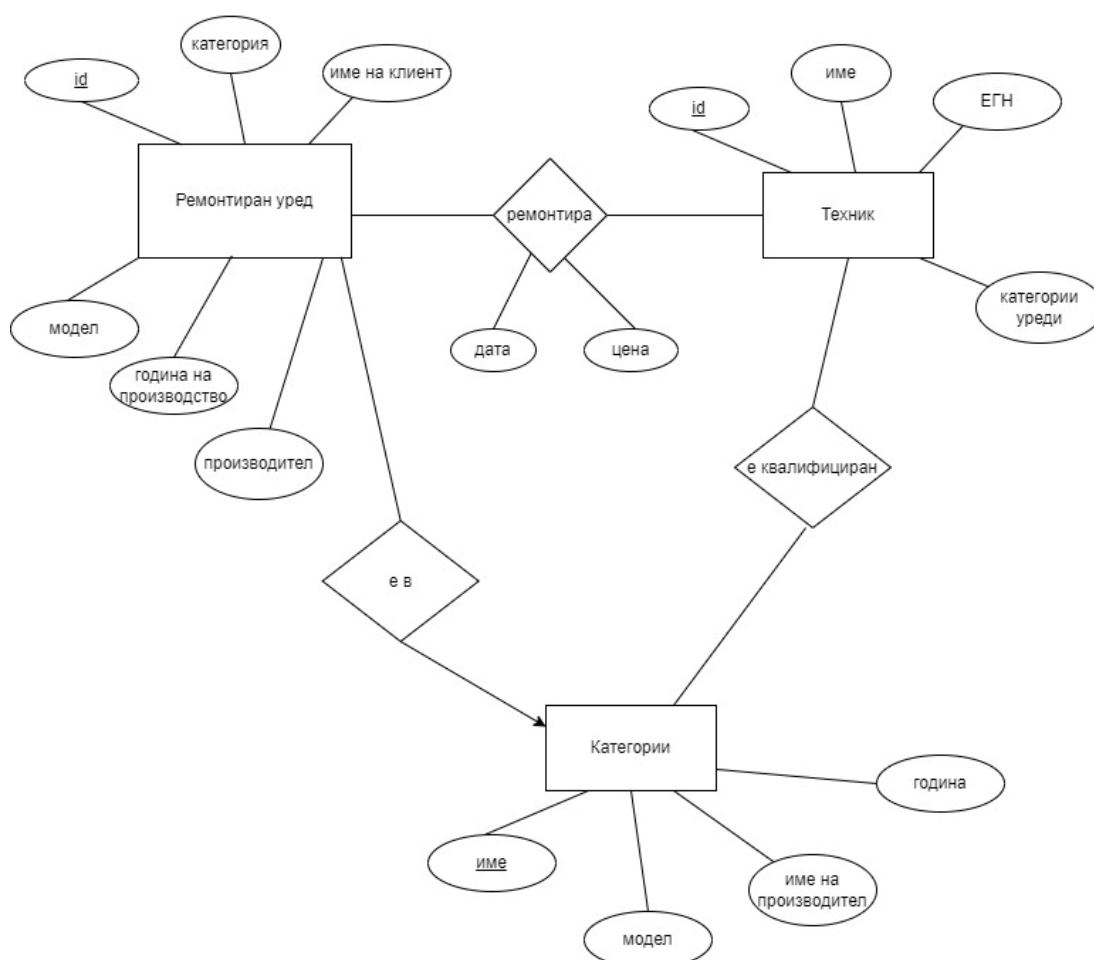
За уред - Годишната на производство (цяло число, по-голямо от 1900)

За техник - Име (до 100 символа)

За всичк положителни числа - проверка за > 0

7. E/R модел на данни

Е/Р модел на база от данни за сервиз за битова техника (Тема 5)



8. Релационен модел на данни

- Преобразуване на множествата от същности:

Ремонтиран уред(уникален идентификатор, категория, производител, модел, име на клиент, година на производство)

Техник(уникален идентификатор, име, ЕГН, категории уреди)

- Преобразуване на връзката много-много

Ремонтира(id_уред, id_техник, дата, цена)

Е квалифициран(id_техник, име_категория)

- Получаваме следния следния релационен модел

Ремонтиран уред(id, категория, име на клиент, модел, година на производство, име_категория)

Техник(id, име, ЕГН, категории уреди)

Категории(име, модел, име на производител, година)

Ремонтира(id_уред, id_техник, дата, цена)

Е квалифициран(id_техник, име_категория)

- Първични ключове:

Ремонтиран уред:ПК(id)

Техник:ПК(id)

Категории:ПК(име)

Ремонтира:ПК(id_уред, id_техник)

Е квалифициран: ПК(id_техник, име)

- Външни ключове

Ремонтиран уред: FK(име_категория) -> Категории(име)

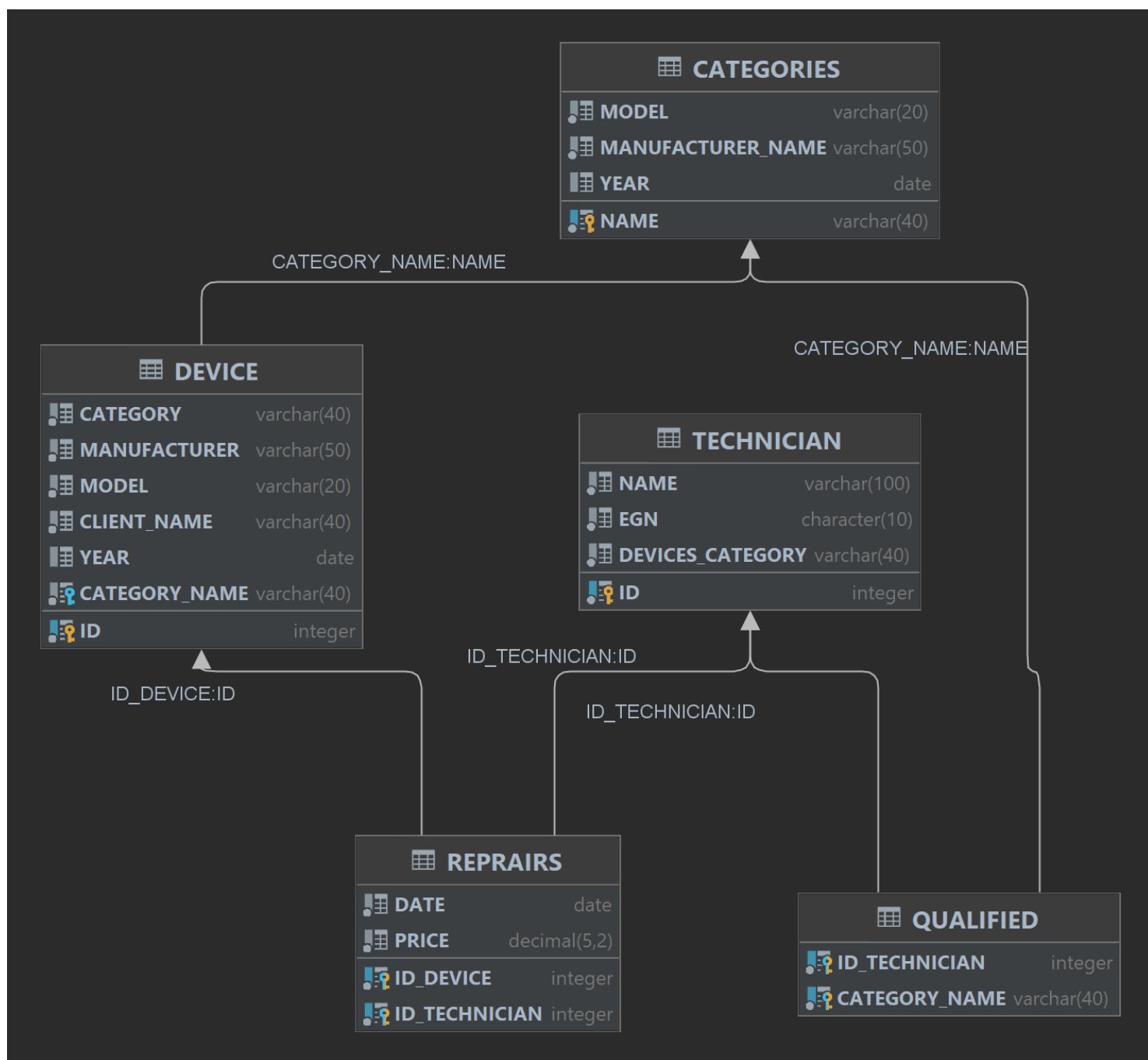
Ремонтира: FK(id_уред) -> Ремонтиран уред(id)

Ремонтира: FK(id_техник) -> Техник(id)

Е квалифициран: FK(id_техник) -> Техник(id)

Е квалифициран: FK(име_категория) -> Категории(име)

9. Схема на базата от данни



10.Изгледы

```
CREATE VIEW TECHNICIAN_REPAIRS_INFO
AS
SELECT T.NAME, T.EGN, R.DATE, R.PRICE
FROM TECHNICIAN T, REPAIRS R
WHERE T.ID = R.ID_TECHNICIAN;

SELECT * FROM TECHNICIAN_REPAIRS_INFO
WHERE PRICE >40;

-- updatable

CREATE VIEW V_CATEGORIES_ALL
AS
SELECT NAME,MODEL, MANUFACTURER_NAME, YEAR
FROM CATEGORIES
WHERE YEAR > '2019-01-01';

SELECT * FROM V_CATEGORIES_ALL;

INSERT INTO V_CATEGORIES_ALL(NAME,MODEL, MANUFACTURER_NAME, YEAR)
VALUES ('Furniture','Bosch', 'Whirlpool WR23MMMS', '2022-03-02');

-- check option

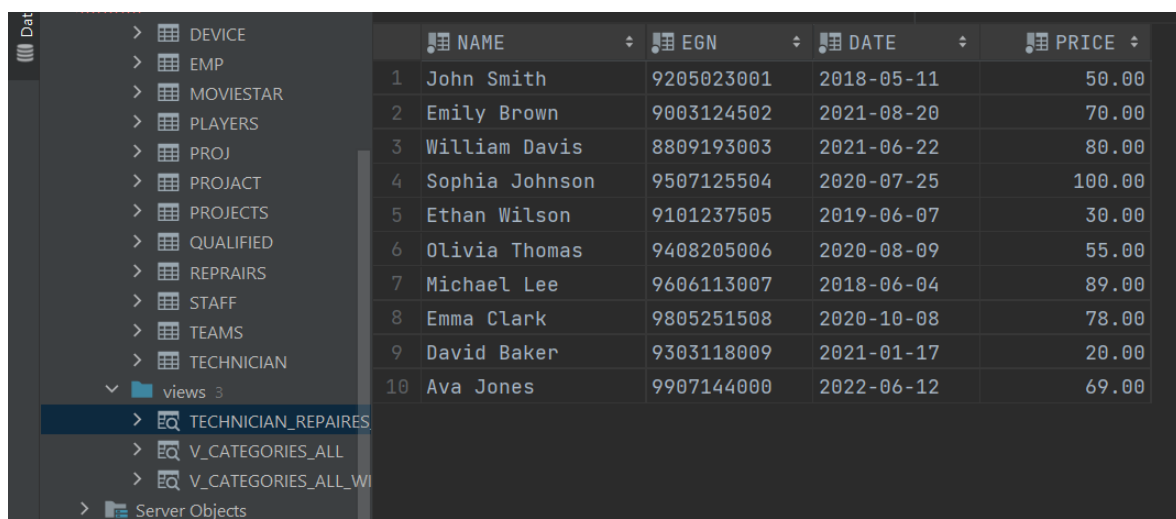
CREATE VIEW V_CATEGORIES_ALL_WITH_CK
AS
SELECT NAME,MODEL, MANUFACTURER_NAME, YEAR
FROM CATEGORIES
WHERE YEAR > '2019-01-01'
WITH CHECK OPTION ;

SELECT * FROM V_CATEGORIES_ALL_WITH_CK;

INSERT INTO V_CATEGORIES_ALL_WITH_CK(NAME, MODEL, MANUFACTURER_NAME, YEAR)
VALUES ('Sink','Frigidaire', 'MI1 M234','2021-04-30'); -- OK

SELECT * FROM V_CATEGORIES_ALL_WITH_CK;

INSERT INTO V_CATEGORIES_ALL_WITH_CK(NAME, MODEL, MANUFACTURER_NAME, YEAR)
VALUES ('Boiler','Tefal', 'Tefal 234L','2015-04-30'); -- NOT OK
```

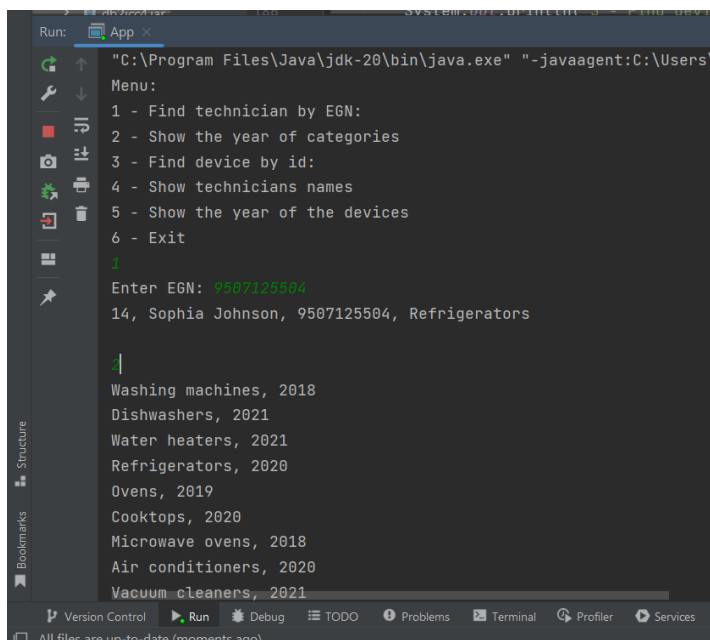


	NAME	EGN	DATE	PRICE
1	John Smith	9205023001	2018-05-11	50.00
2	Emily Brown	9003124502	2021-08-20	70.00
3	William Davis	8809193003	2021-06-22	80.00
4	Sophia Johnson	9507125504	2020-07-25	100.00
5	Ethan Wilson	9101237505	2019-06-07	30.00
6	Olivia Thomas	9408205006	2020-08-09	55.00
7	Michael Lee	9606113007	2018-06-04	89.00
8	Emma Clark	9805251508	2020-10-08	78.00
9	David Baker	9303118009	2021-01-17	20.00
10	Ava Jones	9907144000	2022-06-12	69.00

	NAME	MODEL	MANUFACTURER_NAME	YEAR
1	Dishwashers	LG	LG LDF5545ST dishwas	2021-08-20
2	Water heaters	Bosch	Bosch ES4 water heat	2021-06-22
3	Refrigerators	Whirlpool	Whirlpool WRX735SDHZ	2020-07-25
4	Ovens	GE	GE JB645RKSS oven	2019-06-07
5	Cooktops	Maytag	Maytag MGC7536DS coo	2020-08-09
6	Air conditioners	Kenmore	Kenmore 81414 air co	2020-10-08
7	Vacuum cleaners	KitchenAid	KitchenAid KVWB606DS	2021-01-17
8	Dryers	Frigidaire	Frigidaire FFRE0633U	2022-06-12
9	Furniture	Bosch	Whirlpool WR23MMMS	2022-03-02
10	Sink	Frigidaire	MI1 M234	2021-04-30

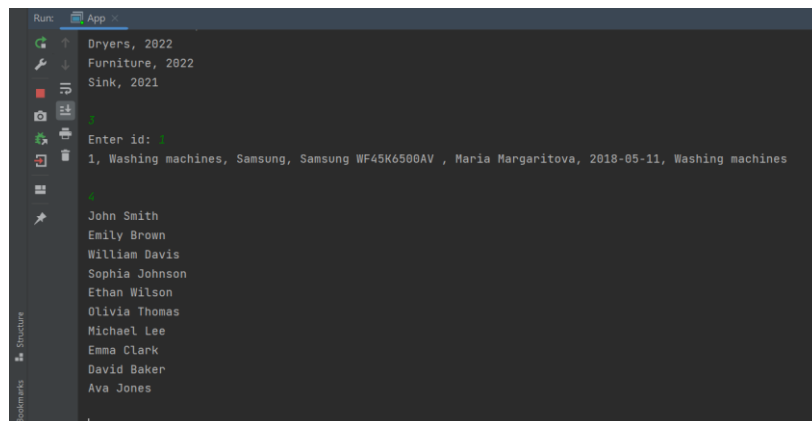
	NAME	MODEL	MANUFACTURER_NAME	YEAR
1	Dishwashers	LG	LG LDF5545ST dishwas	2021-08-20
2	Water heaters	Bosch	Bosch ES4 water heat	2021-06-22
3	Refrigerators	Whirlpool	Whirlpool WRX735SDHZ	2020-07-25
4	Ovens	GE	GE JB645RKSS oven	2019-06-07
5	Cooktops	Maytag	Maytag MGC7536DS coo	2020-08-09
6	Air conditioners	Kenmore	Kenmore 81414 air co	2020-10-08
7	Vacuum cleaners	KitchenAid	KitchenAid KVWB606DS	2021-01-17
8	Dryers	Frigidaire	Frigidaire FFRE0633U	2022-06-12
9	Furniture	Bosch	Whirlpool WR23MMMS	2022-03-02
10	Sink	Frigidaire	MI1 M234	2021-04-30

11. Приложение за достъп до базата



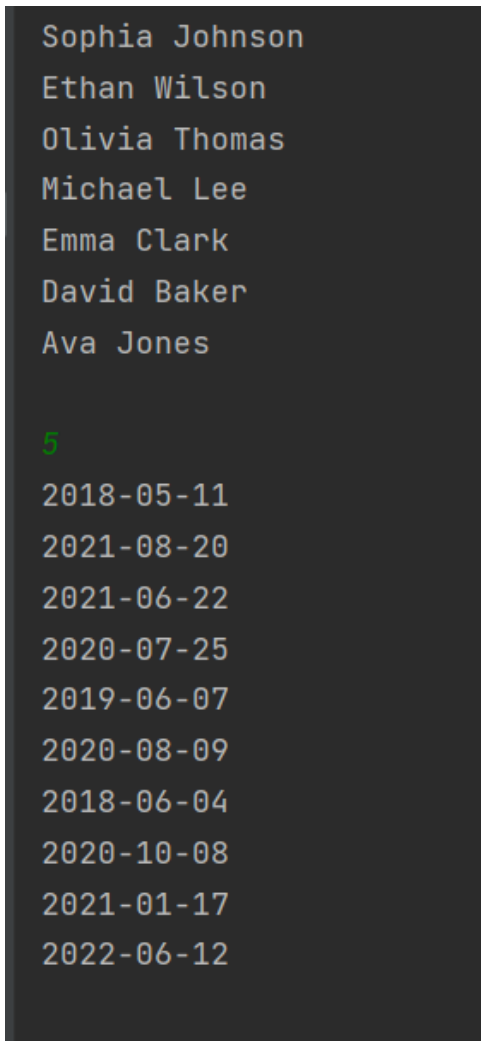
```
Run: App x
"C:\Program Files\Java\jdk-20\bin\java.exe" "-javaagent:C:\Users\
Menu:
1 - Find technician by EGN:
2 - Show the year of categories
3 - Find device by id:
4 - Show technicians names
5 - Show the year of the devices
6 - Exit
Enter EGN: 9507125504
14, Sophia Johnson, 9507125504, Refrigerators

Washing machines, 2018
Dishwashers, 2021
Water heaters, 2021
Refrigerators, 2020
Ovens, 2019
Cooktops, 2020
Microwave ovens, 2018
Air conditioners, 2020
Vacuum cleaners, 2021
```



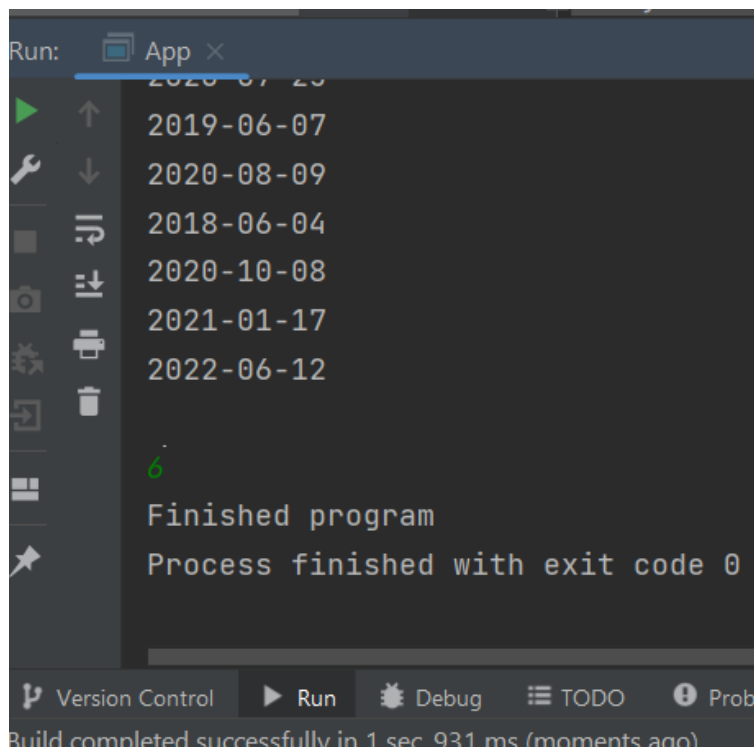
```
Run: App x
Driers, 2022
Furniture, 2022
Sink, 2021
Enter id: 1
1, Washing machines, Samsung, Samsung WF45K6500AV , Maria Margaritova, 2018-05-11, Washing machines

John Smith
Emily Brown
William Davis
Sophia Johnson
Ethan Wilson
Olivia Thomas
Michael Lee
Emma Clark
David Baker
Ava Jones
```



```
Sophia Johnson
Ethan Wilson
Olivia Thomas
Michael Lee
Emma Clark
David Baker
Ava Jones

5
2018-05-11
2021-08-20
2021-06-22
2020-07-25
2019-06-07
2020-08-09
2018-06-04
2020-10-08
2021-01-17
2022-06-12
```



```
Run: App x
2020-07-25
2019-06-07
2020-08-09
2018-06-04
2020-10-08
2021-01-17
2022-06-12

Finished program
Process finished with exit code 0

Version Control Run Debug TODO Prob
Build completed successfully in 1 sec 931 ms (moments ago)
```

```
1 import java.sql.*;
2 import java.util.Scanner;
3
4
5 public class App {
6
7     private static Scanner input = new Scanner(System.in);
8
9     private Connection connection;
10
11     private Statement statement;
12
13     private ResultSet resultSet;
14
15     public void openConnection() {
16
17         try {
18
19             DriverManager.registerDriver(new com.ibm.db2.jcc.DB2Driver());
20
21         } catch (Exception cnfex) {
22
23             System.out.println("Problem in loading or registering IBM DB2 JDBC driver");
24
25         }
26     }
27 }
```

```
16
17         DriverManager.registerDriver(new com.ibm.db2.jcc.DB2Driver());
18
19     } catch (Exception cnfex) {
20
21         System.out.println("Problem in loading or registering IBM DB2 JDBC driver");
22
23         cnfex.printStackTrace();
24
25     }
26
27     try {
28
29         connection = DriverManager.getConnection("jdbc:db2://62.44.108.24:50000/SAMPLE", user: "db2admin", password: "db2admin");
30
31         statement = connection.createStatement();
32
33     } catch (SQLException s) {
34
35         s.printStackTrace();
36
37     }
38
39
40     public void closeConnection() {
41
42         try {
43
44             if (null != connection) {
45
46                 resultSet.close();
47                 statement.close();
48                 connection.close();
49
50             }
51
52         } catch (SQLException s) {
53
54             s.printStackTrace();
55
56         }
57
58     }
59
60     public void select(String stmtnt, int column) {
61
62         try {
63
64             if (null != connection) {
65
66                 resultSet = statement.executeQuery(stmtnt);
67
68                 while (resultSet.next()) {
69
70                     for (int i = 1; i <= column; i++) {
71
72                         System.out.print(resultSet.getString(i) + " ");
73
74                     }
75
76                     System.out.println();
77
78                 }
79
80             }
81
82         } catch (SQLException s) {
83
84             s.printStackTrace();
85
86         }
87
88     }
89 }
```

```
34         s.printStackTrace();
35
36     }
37
38 }
39
40     public void closeConnection() {
41
42         try {
43
44             if (null != connection) {
45
46                 resultSet.close();
47                 statement.close();
48                 connection.close();
49
50             }
51
52         } catch (SQLException s) {
53
54             s.printStackTrace();
55
56         }
57
58     }
59
60     public void select(String stmtnt, int column) {
61
62         try {
63
64             if (null != connection) {
65
66                 resultSet = statement.executeQuery(stmtnt);
67
68                 while (resultSet.next()) {
69
70                     for (int i = 1; i <= column; i++) {
71
72                         System.out.print(resultSet.getString(i) + " ");
73
74                     }
75
76                     System.out.println();
77
78                 }
79
80             }
81
82         } catch (SQLException s) {
83
84             s.printStackTrace();
85
86         }
87
88     }
89 }
```



```
public void select(String stmtnt, int column) {  
    try {  
        resultSet = statement.executeQuery(stmtnt);  
        String result = "";  
        while (resultSet.next()) {  
            for (int i = 1; i <= column; i++) {  
                result += resultSet.getString(i);  
                if (i == column) result += " \n";  
                else result += ", ";  
            }  
            System.out.println(result);  
        } catch (SQLException s) {  
            s.printStackTrace();  
        }  
    }  
    no usages  
    public void insert(String stmtnt) {  
    }  
}
```

```
    try {  
        statement.executeUpdate(stmtnt);  
    } catch (SQLException s) {  
        s.printStackTrace();  
    }  
    System.out.println("Successfully inserted!");  
}  
  
no usages  
public void delete(String stmtnt) {  
    try {  
        statement.executeUpdate(stmtnt);  
    } catch (SQLException s) {  
        s.printStackTrace();  
    }  
}
```

```
private static void choice(App db2obj) {  
    do {  
        int number;  
        number = input.nextInt();  
        switch (number) {  
            case 1:  
                choice1(db2obj);  
                break;  
            case 2:  
                choice2(db2obj);  
                break;  
            case 3:  
                choice3(db2obj);  
                break;  
            case 4:  
                choice4(db2obj);  
                break;  
            case 5:  
                choice5(db2obj);  
                break;  
            case 6:  
                System.out.print("Finished program");  
                return;  
        }  
    }  
}
```

```
File Edit View Navigate Code Refactor Build Run Tools VCS Window Help project [D:\Университет\4. Четвертый семестр\Практикум БД\ЗАДАЧА\И7. Задана\project] - App.java
project src App openConnection
App.java
140 return;
141 default:
142     System.out.println("Invalid choice");
143 }
144 } while (true);
145 }
146
147 @ 1 usage
148 private static void choice1(App db2obj) {
149     System.out.print("Enter EGN: ");
150     String egn = input.next();
151
152     String statement = "SELECT * FROM FN1MI0700004.TECHNICIAN WHERE EGN = " + egn;
153     db2obj.select(statement, column: 4);
154 }
155
156 @ 1 usage
157 private static void choice2(App db2obj) {
158     String statement = "SELECT NAME, YEAR(YEAR) FROM FN1MI0700004.CATEGORIES";
159     db2obj.select(statement, column: 2);
160 }
161
162 @ 1 usage
163 private static void choice3(App db2obj) {
164     System.out.print("Enter id: ");
165     Integer id = input.nextInt();
166
167     String statement = "SELECT * FROM FN1MI0700004.DEVICE WHERE ID = " + id;
```

```
File Edit View Navigate Code Refactor Build Run Tools VCS Window Help project [D:\Университет\4. Четвертый семестр\Практикум БД\ЗАДАЧА\И7. Задана\project] - App.java
project src App openConnection
App.java
161 System.out.print("Enter id: ");
162 Integer id = input.nextInt();
163
164 String statement = "SELECT * FROM FN1MI0700004.DEVICE WHERE ID = " + id;
165 db2obj.select(statement, column: 7);
166 }
167
168 @ 1 usage
169 private static void choice4(App db2obj) {
170     String statement = "SELECT NAME FROM FN1MI0700004.TECHNICIAN";
171     db2obj.select(statement, column: 1);
172 }
173
174 @ 1 usage
175 private static void choice5(App db2obj) {
176     String statement = "SELECT YEAR FROM FN1MI0700004.DEVICE";
177     db2obj.select(statement, column: 1);
178 }
179
180 no usages
181 public static void main(String[] args) {
182
183     App db2obj = new App();
184     db2obj.openConnection();
185
186     System.out.println("Menu:");
187     System.out.println("1 - Find technician by EGN:");
188 }
```

```
File Edit View Navigate Code Refactor Build Run Tools VCS Window Help project - App.java
project src App statement
App.java
177 no usages
178 public static void main(String[] args) {
179
180     App db2obj = new App();
181     db2obj.openConnection();
182
183     System.out.println("Menu:");
184     System.out.println("1 - Find technician by EGN:");
185     System.out.println("2 - Show the year of categories");
186     System.out.println("3 - Find device by id:");
187     System.out.println("4 - Show technicians names");
188     System.out.println("5 - Show the year of the devices");
189     System.out.println("6 - Exit");
190
191     choice(db2obj);
192
193     db2obj.closeConnection();
194 }
195
196 }
```