

# Hospital Database

**Prepared by**

Caique Bispo Ferreira (301093125)

Chuyun Yang (301038102)

Manal Hagos(301027317)

Meera Mohan (301093014)

**Centennial College**

**Fall Semester**

**December 10 2019**

## Contents

<b>Hospital Database</b>	<b>1</b>
1. Problem domain description	3
2. Create Tables	4
2.1 Company	4
2.2 Patient	5
2.3 Doctor	6
2.4 CompanyXPatient	7
2.5 Prescription	8
3. Create Sequences	9
3.1 Sequence Company	9
3.2 Sequence CompanyXPatient	9
3.3 Sequence Doctor	10
3.4 Sequence Patient	10
3.5 Sequence Prescription	11
4. Create Indexes	12
4.1 Index Company	12
4.2 Index Patient	13
5. ER Diagram	14
6. Functions	15
6.1 Count the number of patients' prescriptions of the company	15
6.2 Count the number of prescriptions of the doctor	16
7. Procedures	17
7.1 Update the number of prescriptions per company	17
7.2 Update the number of prescriptions per doctor	18
8. Trigger – Automatic update the number of prescriptions	19
9. Interface	19
10. Web Application	21

## 1. Problem domain description

The purpose of this project is to organize, control and manage the patients' files in order to provide a general perspective when doctors need to check any historic medical information. These personal data allow not only doctors to be more accurate in their prescription, but also provide a general overview of the patients' life in hospitals or/and clinics. Moreover, insurance companies can also use this data to provide cheaper health insurance plans.

## 2. Create Tables

### 2.1 Company

Worksheet Query Builder

```
1 CREATE TABLE COMPANY /*HEALTH_INSURANCE*/
2 (
3     COMPANY_ID    NUMBER(5) ,
4     NAME          VARCHAR2(15),
5     CONTACT_FNAME VARCHAR2(15),
6     CONTACT_LNAME VARCHAR2(15),
7     PHONE         NUMBER(12),
8     PRESC_AMOUNT  NUMBER,
9
10    CONSTRAINT COMPANY_ID_PK PRIMARY KEY(COMPANY_ID)
11 );
12
```

Script Output x

Task completed in 0.135 seconds

Table COMPANY created.

Worksheet Query Builder

```
63 VALUES (COMPANY_COMPANY_ID_SEQ.NEXTVAL,'Manaly',
64     'Meme',
65     'Poland',
66     5558099,
67     0);
68 INSERT INTO COMPANY(COMPANY_ID,NAME,CONTACT_FNAME,CONTACT_LNAME,PHONE,PRESA_AMOUNT)
69 VALUES (COMPANY_COMPANY_ID_SEQ.NEXTVAL,'Ayman',
70     'Amalia',
71     'Oglo',
72     5558034,
73     0);
74 SELECT * FROM COMPANY;
```

Query Result x

SQL | All Rows Fetched: 11 in 0.036 seconds

	COMPANY_ID	NAME	CONTACT_FNAME	CONTACT_LNAME	PHONE	PRESA_AMOUNT
1	1	Prundetia	Jan	Johnson	5554567	0
2	2	State Farm	Sam	Smith	5558765	0
3	3	Nationwide	Fandy	Buck	5559854	0
4	4	Sunshine	Eduardo	Jozeph	5558711	0
5	5	Sunshine	ALEXANDRA	DAOOD	5558720	0
6	6	Sunset	Ahmed	Seraj	5558350	0
7	7	Medgulf	Julia	Mathio	5558512	0
8	8	Bupa	Maria	Sultan	5558507	0
9	9	Admas	Joana	Adam	5558000	0
10	10	Manaly	Meme	Poland	5558099	0
11	11	Ayman	Amalia	Oglo	5558034	0

## 2.2 Patient

Worksheet Query Builder

```
1 CREATE TABLE PATIENT |
2 (
3     PATIENT_ID NUMBER(5),
4     FNAME  VARCHAR2 (15) NOT NULL,
5     LNAME  VARCHAR2(15) NOT NULL,
6     STREET VARCHAR2(15) NOT NULL,
7     CITY   VARCHAR2 (10) NOT NULL,
8     STATE  VARCHAR2 (12) NOT NULL,
9     ZIP    VARCHAR2(5) NOT NULL,
10    PHONE  NUMBER(12),
11
12    CONSTRAINT PATIENT_ID_PK PRIMARY KEY(PATIENT_ID)
13 );
14
```

Script Output x

Task completed in 0.112 seconds

Table PATIENT created.

```
102
103 INSERT INTO PATIENT(PATIENT_ID,FNAME,LNAME,STREET,CITY,STATE,ZIP,PHONE)
104 VALUES (PATIENT_PATIENT_ID_SEQ.NEXTVAL,
105     'Amany',
106     'Abdalmohsen',
107     'Leside Ave',
108     'London',
109     'ON',
110     45710,
111     6863382);
112 COMMIT;
113 SELECT * FROM PATIENT;
```

Script Output x Query Result x

All Rows Fetched: 11 in 0.029 seconds

	PATIENT_ID	FNAME	LNAME	STREET	CITY	STATE	ZIP	PHONE
1	1	Jim	Perotti	12 Oak St	Shade	OH	45701	7863333
2	2	Glien	Stamier	1 Walnut Ave	Athens	FL	45702	7863333
3	3	Maddy	Gary	45 marple St	Toronto	ON	45703	68633344
4	4	Black	Shelton	3 Ekn Oliv	Shade	OH	45704	5863333
5	5	EJ	Thomson	92 Ash St	Orlando	FL	85705	5863332
6	6	Dyrk	Ashton	33 Poptar Ave	Columbus	OH	45706	6863332
7	7	Neema	Saeed	Steeles Ave	Markham	ON	45707	6863330
8	8	Neema	Saeed	Steeles Ave	Markham	ON	45707	6863330
9	9	Sofia	Shah	Cosburn	Toronto	ON	45708	6863344
10	10	Eman	Mohamed	Clock Tower Ave	Ajax	ON	45709	6863319
11	11	Amany	Abdalmohsen	Leside Ave	London	ON	45710	6863382

## 2.3 Doctor

Worksheet Query Builder

```
1 CREATE TABLE DOCTOR
2 (
3     DOCTOR_ID NUMBER(5),
4     FNAME VARCHAR2(15) NOT NULL,
5     LNAME VARCHAR2(15) NOT NULL,
6     PHONE NUMBER(12),
7     BEEPER VARCHAR2(12),
8     PRESC_AMOUNT NUMBER,
9
10    CONSTRAINT DOCTOR_ID_PK PRIMARY KEY(DOCTOR_ID)
11 );
```

Script Output x

Task completed in 0.151 seconds

Table DOCTOR created.

Worksheet Query Builder

```
162 INSERT INTO DOCTOR(DOCTOR_ID,FNAME,LNAME,PHONE,BEEPER,PRESC_AMOUNT)
163 VALUES (DOCTOR_DOCTOR_ID_SEQ.NEXTVAL,
164         'Shana',
165         'Khlar',
166         647564088,
167         5551222,
168         0);
169
170 SELECT * FROM DOCTOR;
```

Script Output x Query Result x

SQL | All Rows Fetched: 13 in 0.032 seconds

	DOCTOR_ID	FNAME	LNAME	PHONE	BEEPER	PRESC_AMOUNT
1	1	Alex	JACKSON	6477622262	5551212	0
2	2	perry	cox	6477622200	5551213	0
3	3	Eloit	Ried	6477622211	5551214	0
4	4	Jp	Dorain	6477622233	5551215	0
5	5	Bob	Kelso	6477622244	5551216	0
6	6	JENI	AGA	6477633334	5551217	0
7	7	ELLISA	FAHAD	6477633300	5551218	0
8	8	ROBERT	ARDERSON	6477633312	5551219	0
9	9	OLIVER	NELSON	6477633331	5551218	0
10	10	IDRIS	HAGOS	6477633339	5551219	0
11	11	Bella	Khan	647763323	5551220	0
12	12	Alisandra	Khosie	647764000	5551221	0
13	13	Shana	Khlar	647564088	5551222	0



## 2.4 CompanyXPatient

Worksheet Query Builder

```
1 CREATE TABLE COMPANYXPATIENT
2 (
3   ID NUMBER(5),
4   COMPANY_ID NUMBER(5),
5   PATIENT_ID NUMBER(5),
6
7   CONSTRAINT COMPANYXPATIENT_ID_PK PRIMARY KEY(ID),
8   CONSTRAINT COMPANYXPATIENT_UNIQUE UNIQUE(COMPANY_ID,PATIENT_ID),
9   CONSTRAINT PATIENT_ID_FK FOREIGN KEY(PATIENT_ID) REFERENCES PATIENT(PATIENT_ID),
10  CONSTRAINT COMPANY_ID_FK FOREIGN KEY(COMPANY_ID) REFERENCES COMPANY(COMPANY_ID)
11 );
12
```

Script Output x

Task completed in 0.144 seconds

Table COMPANYXPATIENT created.

Worksheet Query Builder

```
27
28 INSERT INTO COMPANYXPATIENT(ID,COMPANY_ID,PATIENT_ID)
29 VALUES (COMPANYXPATIENT_ID_SEQ.NEXTVAL,2,1);
30
31 INSERT INTO COMPANYXPATIENT(ID,COMPANY_ID,PATIENT_ID)
32 VALUES (COMPANYXPATIENT_ID_SEQ.NEXTVAL,3,1);
33
34 INSERT INTO COMPANYXPATIENT(ID,COMPANY_ID,PATIENT_ID)
35 VALUES (COMPANYXPATIENT_ID_SEQ.NEXTVAL,3,2);
36 COMMIT;
37 SELECT * FROM COMPANYXPATIENT;
```

Script Output x Query Result x

SQL | All Rows Fetched: 11 in 0.024 seconds

	ID	COMPANY_ID	PATIENT_ID
1	1	1	1
2	2	1	2
3	3	2	3
4	4	2	4
5	5	3	5
6	6	3	6
7	7	3	3
8	8	3	4
9	9	2	1
10	10	3	1
11	11	3	2

## 2.5 Prescription

Worksheet Query Builder

```
1 CREATE TABLE PRESCRIPTION
2 (
3     PRESC_ID NUMBER(5),
4     ID        NUMBER(5),
5     DOCTOR_ID NUMBER(5),
6     PDATE     DATE NOT NULL,
7     POBS      VARCHAR2(50) NOT NULL,
8
9     CONSTRAINT PRESC_ID_PK PRIMARY KEY(PRESC_ID),
10    CONSTRAINT COMPANYXPATIENT_ID_FK FOREIGN KEY(ID) REFERENCES COMPANYXPATIENT(ID),
11    CONSTRAINT DOCTOR_ID_FK FOREIGN KEY(DOCTOR_ID) REFERENCES DOCTOR(DOCTOR_ID)
12 );
```

Script Output x

Task completed in 0.166 seconds

Table PRESCRIPTION created.

Worksheet Query Builder

```
38
39 INSERT INTO PRESCRIPTION(PRESC_ID,ID,DOCTOR_ID,PDATE,POBS)
40 VALUES (PRESCRIPTION_PRESC_ID_SEQ.NEXTVAL, 7,5,
41         TO_DATE('19-04-3', 'YY-MM-DD'),'FEFADOL');
42
43 INSERT INTO PRESCRIPTION(PRESC_ID,ID,DOCTOR_ID,PDATE,POBS)
44 VALUES (PRESCRIPTION_PRESC_ID_SEQ.NEXTVAL, 6,5,
45         TO_DATE('19-06-5', 'YY-MM-DD'),'TORADOL ');
46
47 COMMIT;
48 SELECT * FROM PRESCRIPTION;
49
```

Script Output x Query Result x

All Rows Fetched: 11 in 0.028 seconds

	PRESC_ID	ID	DOCTOR_ID	PDATE	POBS
1	1	1	1	13-APR-19	Acetaminophen
2	2	2	2	13-JUN-19	Adderall
3	3	3	3	10-MAR-19	Alprazolam
4	4	3	3	23-FEB-19	Amitriptyline
5	5	4	4	23-FEB-19	Amlodipine
6	6	5	5	20-FEB-19	Amlodipine
7	7	7	7	25-FEB-19	FEFADOL
8	8	6	6	27-FEB-19	FEFADOL
9	9	6	6	28-FEB-19	Panadol
10	10	7	7	03-APR-19	FEFADOL
11	11	6	6	05-JUN-19	TORADOL



### 3. Create Sequences

#### 3.1 Sequence Company

The screenshot shows the SQL Developer Query Builder interface. The SQL script is as follows:

```
1 CREATE INDEX COMPANY_NAME_I
2 ON COMPANY(NAME);
3
4 CREATE SEQUENCE COMPANY_COMPANY_ID_SEQ;
5
6 INSERT INTO COMPANY(COMPANY_ID,NAME,CONTACT_FNAME,CONTACT_LNAME,PHONE,PRESO_AMOUNT)
7 VALUES (COMPANY_COMPANY_ID_SEQ.NEXTVAL,'Prundetia',
8         'Jan',
9         'Johnson',
10        5554567,
11        0);
12
13 INSERT INTO COMPANY(COMPANY_ID,NAME,CONTACT_FNAME,CONTACT_LNAME,PHONE,PRESO_AMOUNT)
14 VALUES (COMPANY_COMPANY_ID_SEQ.NEXTVAL,'State Farm',
15         'Sam',
```

The Script Output pane shows the following messages:

Index COMPANY\_NAME\_I created.

Sequence COMPANY\_COMPANY\_ID\_SEQ created.

1 row inserted.

#### 3.2 Sequence CompanyXPatient

The screenshot shows the SQL Developer Query Builder interface. The SQL script is as follows:

```
1 CREATE SEQUENCE COMPANYXPATIENT_ID_SEQ;
2
3 INSERT INTO COMPANYXPATIENT(ID,COMPANY_ID,PATIENT_ID)
4 VALUES (COMPANYXPATIENT_ID_SEQ.NEXTVAL,1,1);
5
6 INSERT INTO COMPANYXPATIENT(ID,COMPANY_ID,PATIENT_ID)
7 VALUES (COMPANYXPATIENT_ID_SEQ.NEXTVAL,1,2);
8
9 INSERT INTO COMPANYXPATIENT(ID,COMPANY_ID,PATIENT_ID)
10 VALUES (COMPANYXPATIENT_ID_SEQ.NEXTVAL,2,3);
11
12 INSERT INTO COMPANYXPATIENT(ID,COMPANY_ID,PATIENT_ID)
13 VALUES (COMPANYXPATIENT_ID_SEQ.NEXTVAL,2,4);
14
15
16 INSERT INTO COMPANYXPATIENT(ID,COMPANY_ID,PATIENT_ID)
17 VALUES (COMPANYXPATIENT_ID_SEQ.NEXTVAL,3,5);
```

The Script Output pane shows the following messages:

Sequence COMPANYXPATIENT\_ID\_SEQ created.

1 row inserted.

### 3.3 Sequence Doctor

```
Worksheet | Query Builder
1 CREATE SEQUENCE DOCTOR_DOCTOR_ID_SEQ;
2
3 INSERT INTO DOCTOR(DOCTOR_ID,FNAME,LNAME,PHONE,BEEPER,PRESCH_AMOUNT)
4 VALUES (DOCTOR_DOCTOR_ID_SEQ.NEXTVAL,
5 'ALex',
6 'JACKSON ',
7 6477622262,
8 5551212,
9 0);
10
11 INSERT INTO DOCTOR(DOCTOR_ID,FNAME,LNAME,PHONE,BEEPER,PRESCH_AMOUNT)
12 VALUES (DOCTOR_DOCTOR_ID_SEQ.NEXTVAL,
13 'perry',
14 'cox ',
15 6477622200,
16 5551213,
17 0);
18
```

Script Output x

Task completed in 1.207 seconds

1 row inserted.

### 3.4 Sequence Patient

```
Worksheet | Query Builder
1 CREATE INDEX PATIENT_LAST_NAME_I
2 ON PATIENT(LNAME);
3
4 CREATE SEQUENCE PATIENT_PATIENT_ID_SEQ;
5
6 INSERT INTO PATIENT(PATIENT_ID,FNAME,LNAME,STREET,CITY,STATE,ZIP,PHONE)
7 VALUES (PATIENT_PATIENT_ID_SEQ.NEXTVAL,
8 'Jim',
9 'Perotti',
10 '12 Oak St',
11 'Shade',
12 'OH',
13 45701,
14 7863333);
15
```

Script Output x

Task completed in 1.028 seconds

Index PATIENT\_LAST\_NAME\_I created.

Sequence PATIENT\_PATIENT\_ID\_SEQ created.

1 row inserted.

### 3.5 Sequence Prescription

Worksheet

Query Builder

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

```
CREATE SEQUENCE PRESCRIPTION_PRESC_ID_SEQ;  
  
INSERT INTO PRESCRIPTION(PRESC_ID,DOCTOR_ID,PDATE,POBS)  
VALUES (PRESCRIPTION_PRESC_ID_SEQ.NEXTVAL, 1,1,  
        TO_DATE('19-04-13', 'YY-MM-DD'),'Acetaminophen');  
  
INSERT INTO PRESCRIPTION(PRESC_ID,DOCTOR_ID,PDATE,POBS)  
VALUES (PRESCRIPTION_PRESC_ID_SEQ.NEXTVAL, 2,2,  
        TO_DATE('19-06-13', 'YY-MM-DD'),'Adderall');  
  
INSERT INTO PRESCRIPTION(PRESC_ID,DOCTOR_ID,PDATE,POBS)  
VALUES (PRESCRIPTION_PRESC_ID_SEQ.NEXTVAL, 3,3,  
        TO_DATE('19-03-10', 'YY-MM-DD'),'Alprazolam');  
  
INSERT INTO PRESCRIPTION(PRESC_ID,DOCTOR_ID,PDATE,POBS)  
VALUES (PRESCRIPTION_PRESC_ID_SEQ.NEXTVAL, 3,3,  
        TO_DATE('19-02-23', 'YY-MM-DD'),'Amitriptyline');
```

Script Output x

Query Result x

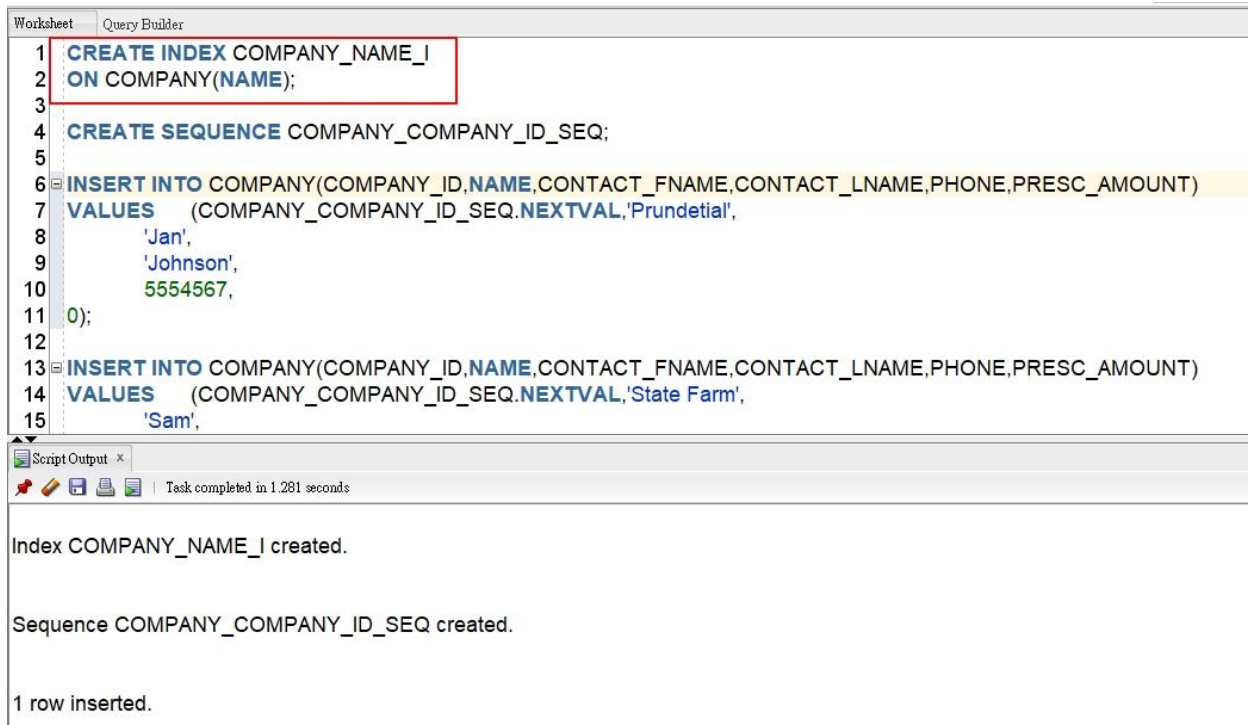
Task completed in 1.02 seconds

Sequence PRESCRIPTION\_PRESC\_ID\_SEQ created.

1 row inserted.

## 4. Create Indexes

### 4.1 Index Company



The screenshot displays a SQL query editor with a 'Worksheet' and 'Query Builder' tab. The SQL script is as follows:

```
1 CREATE INDEX COMPANY_NAME_I
2 ON COMPANY(NAME);
3
4 CREATE SEQUENCE COMPANY_COMPANY_ID_SEQ;
5
6 INSERT INTO COMPANY(COMPANY_ID,NAME,CONTACT_FNAME,CONTACT_LNAME,PHONE,PRESA_AMOUNT)
7 VALUES (COMPANY_COMPANY_ID_SEQ.NEXTVAL,'Prundetia',
8         'Jan',
9         'Johnson',
10        5554567,
11        0);
12
13 INSERT INTO COMPANY(COMPANY_ID,NAME,CONTACT_FNAME,CONTACT_LNAME,PHONE,PRESA_AMOUNT)
14 VALUES (COMPANY_COMPANY_ID_SEQ.NEXTVAL,'State Farm',
15         'Sam',
```

Below the script, the 'Script Output' window shows the results of the execution:

Index COMPANY\_NAME\_I created.

Sequence COMPANY\_COMPANY\_ID\_SEQ created.

1 row inserted.

## 4.2 Index Patient

The screenshot shows a SQL query editor window with a 'Worksheet' tab and a 'Query Builder' tab. The query is as follows:

```
1 CREATE INDEX PATIENT_LAST_NAME_I
2 ON PATIENT(LNAME);
3
4 CREATE SEQUENCE PATIENT_PATIENT_ID_SEQ;
5
6 INSERT INTO PATIENT(PATIENT_ID,FNAME,LNAME,STREET,CITY,STATE,ZIP,PHONE)
7 VALUES (PATIENT_PATIENT_ID_SEQ.NEXTVAL,
8         'Jim',
9         'Perotti',
10        '12 Oak St',
11        'Shade',
12        'OH',
13        45701,
14        7863333);
15
```

The first two lines of the query are highlighted with a red box. Below the query editor is a 'Script Output' window showing the results of the execution:

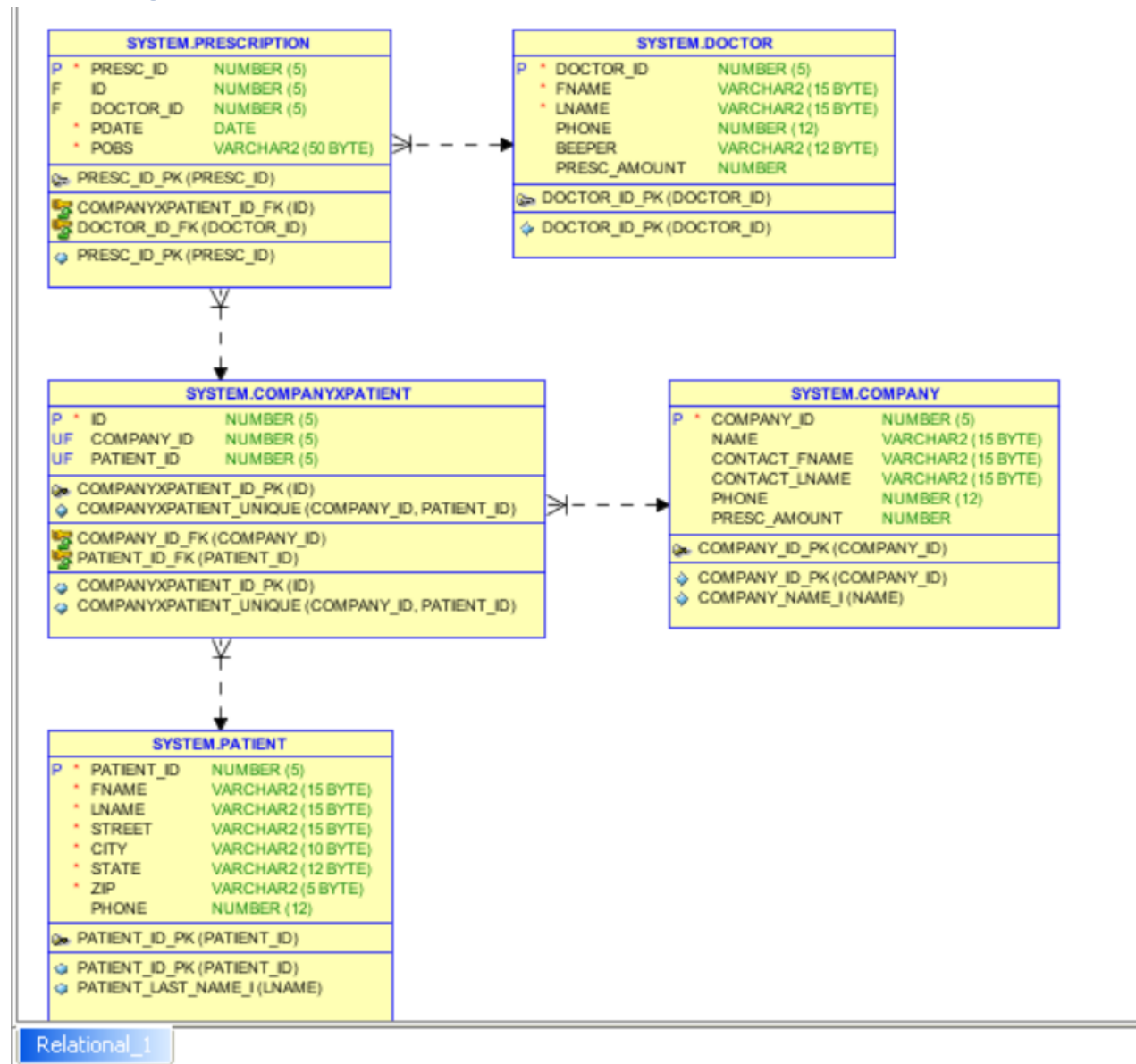
Task completed in 1.028 seconds

Index PATIENT\_LAST\_NAME\_I created.

Sequence PATIENT\_PATIENT\_ID\_SEQ created.

1 row inserted.

## 5. ER Diagram





## 6. Functions

### 6.1 Count the number of patients' prescriptions of the company

Worksheet




Query Builder

```
1 CREATE OR REPLACE FUNCTION CALCTOT_PRESC_COMPANY
2   (vCOMPANY_ID COMPANY.COMPANY_ID%TYPE )
3   RETURN NUMBER
4 IS
5   TOTAL NUMBER := 0;
6 BEGIN
7   SELECT COUNT(*)
8   INTO TOTAL
9   FROM PRESCRIPTION
10  WHERE ID IN (
11    SELECT ID
12    FROM COMPANYXPATIENT
13    WHERE COMPANYXPATIENT.COMPANY_ID = vCOMPANY_ID
14  );
15  RETURN TOTAL;
16 END;
```

DECLARE

```
18 vCOMPANY_ID NUMBER := 1;
19 BEGIN
20   DBMS_OUTPUT.PUT_LINE('COMPANY_ID: '||vCOMPANY_ID||',||
21     TOTAL: '|| CALCTOT_PRESC_COMPANY(vCOMPANY_ID));
22 END;
```




Script Output x

   | Task completed in 0.343 seconds

Function CALCTOT\_PRESC\_COMPANY compiled

PL/SQL procedure successfully completed.

Dbms Output

   | Buffer Size: 20000

yyun x

COMPANY\_ID: 1,TOTAL: 2

## 6.2 Count the number of prescriptions of the doctor

The screenshot displays the Oracle SQL Developer interface with the 'Query Builder' tab active. The main editor contains the following PL/SQL code:

```
1 CREATE OR REPLACE FUNCTION CALCTOT_PRESC_DOCTOR(IN_DOCTOR_ID IN NUMBER)
2 RETURN NUMBER
3 IS
4     PRESCRIPTION_DETAILS NUMBER;
5 BEGIN
6     SELECT COUNT(*)
7     INTO PRESCRIPTION_DETAILS
8     FROM PRESCRIPTION PRESCRIPTION
9     WHERE PRESCRIPTION.DOCTOR_ID = IN_DOCTOR_ID;
10
11     RETURN( PRESCRIPTION_DETAILS );
12 END;
13
14 DECLARE
15     vDOCTOR_ID NUMBER := 1;
16 BEGIN
17     DBMS_OUTPUT.PUT_LINE( 'DOCTOR_ID: '||vDOCTOR_ID||','||
18         'TOTAL: '|| CALCTOT_PRESC_DOCTOR(vDOCTOR_ID));
19 END;
```

Below the code editor, the 'Script Output' window shows the execution results:

Task completed in 0.394 seconds

Function CALCTOT\_PRESC\_DOCTOR compiled

PL/SQL procedure successfully completed.

The 'Dbms Output' window at the bottom shows the output of the execution:

DOCTOR\_ID: 1,TOTAL: 1

## 7. Procedures

### 7.1 Update the number of prescriptions per company

The screenshot displays the Oracle SQL Developer environment. The top pane shows a PL/SQL procedure named `UPD_TOT_PRESC_COMPANY` being created or replaced. The procedure logic includes a variable `OLD_PRESC_AMOUNT`, a record type `type_COMPANY_PRESC`, a cursor `C_COMPANY_PRESC`, and a `SELECT` statement that calculates the total prescriptions per company using `CALCTOT_PRESC_COMPANY`.

The middle pane, titled "Script Output", shows the message "Task completed in 0.352 seconds".

The bottom pane, titled "Dbms Output", shows the message "PL/SQL procedure successfully completed."

The bottom-most pane, titled "yun", displays the output of the procedure, showing the old and new prescription amounts for 11 companies:

COMPANY_ID	OLD_PRESC_AMOUNT	NEW_PRESC_AMOUNT
1	2	2
2	3	3
3	6	6
4	0	0
5	0	0
6	0	0
7	0	0
8	0	0
9	0	0
10	0	0
11	0	0

## 7.2 Update the number of prescriptions per doctor

The screenshot displays the SQL Developer interface with the Query Builder tab active. The SQL editor contains the following code:

```
21 CREATE OR REPLACE
22 PROCEDURE UPD_TOT_PRESC_DOCTOR
23 IS
24     vOld_PRESC_AMOUNT DOCTOR.PRESC_AMOUNT%TYPE := 0;
25     TYPE type_DOCTOR_PRESC IS RECORD (
26         DOCTOR_ID DOCTOR.DOCTOR_ID%TYPE,
27         TOTAL NUMBER
28     );
29     rec_DOCTOR_PRESC type_DOCTOR_PRESC;
30     CURSOR C_DOCTOR_PRESC IS
31         SELECT *
32         FROM (
33             SELECT DOCTOR_ID,
34                 CALCOTOT_PRESC_DOCTOR(DOCTOR_ID) AS TOTAL
35             FROM DOCTOR
```

Below the SQL editor, the Script Output window shows the execution results:

Task completed in 0.38 seconds

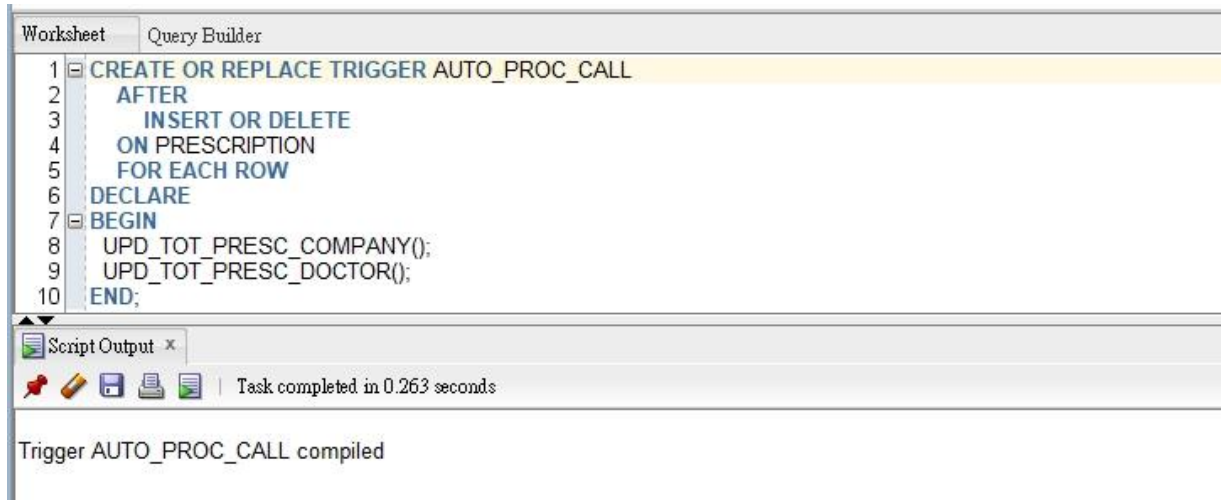
Procedure UPD\_TOT\_PRESC\_DOCTOR compiled

PL/SQL procedure successfully completed.

The Dbms Output window shows the following data:

DOCTOR_ID	OLD_PRESC_AMOUNT	NEW_PRESC_AMOUNT
1	0	1
2	0	1
3	0	2
4	0	2
5	0	3
6	0	1
7	0	0
8	0	1
9	0	0
10	0	0
11	0	0
12	0	0
13	0	0

## 8. Trigger – Automatic update the number of prescriptions



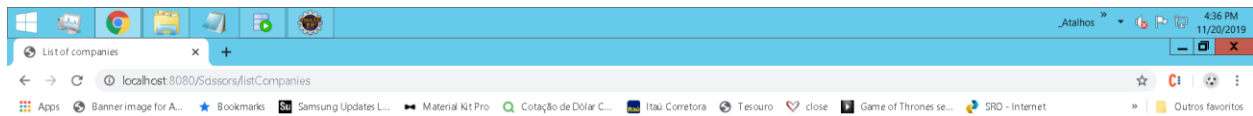
```
1 CREATE OR REPLACE TRIGGER AUTO_PROC_CALL
2 AFTER
3   INSERT OR DELETE
4   ON PRESCRIPTION
5   FOR EACH ROW
6 DECLARE
7 BEGIN
8   UPD_TOT_PRESC_COMPANY();
9   UPD_TOT_PRESC_DOCTOR();
10 END;
```

Script Output x

Task completed in 0.263 seconds

Trigger AUTO\_PROC\_CALL compiled

## 9. Interface



Companies -> List of companies

[Call procedure](#)

Name	Contact's First Name	Contact's Last Name	Phone	Prescription's Amount
Prundetal	Jan	Johnson	5554567	0
State Farm	Sam	Smith	5558765	0
Nationwide	Fandy	Buck	5559854	0

Click in call procedure



Companies -> List of companies

[Call procedure](#)

Name	Contact's First Name	Contact's Last Name	Phone	Prescription's Amount
Prundetal	Jan	Johnson	5554567	2
State Farm	Sam	Smith	5558765	3
Nationwide	Fandy	Buck	5559854	1

The screenshot shows the Eclipse IDE interface with a workspace named 'workspace - Eclipse IDE'. The console window displays the following logs:

```

:: Spring Boot ::                (v2.1.3.RELEASE)

2019-11-20 16:36:06.439 INFO 3760 --- [main] com.scissors.auth.WebApplication : Starting WebApplication v5.3.6
2019-11-20 16:36:06.439 INFO 3760 --- [main] com.scissors.auth.WebApplication : No active profile set, falling
2019-11-20 16:36:09.032 INFO 3760 --- [main] o.a.c.c.C.[.localhost].[/Scissors] : Initializing Spring embedded W
2019-11-20 16:36:10.017 INFO 3760 --- [main] o.hibernate.jpa.internal.util.LogHelper : HHH000204: Processing Persiste
name: default
...
2019-11-20 16:36:10.203 INFO 3760 --- [main] org.hibernate.Version : HHH000412: Hibernate Core {5.3
2019-11-20 16:36:10.209 INFO 3760 --- [main] org.hibernate.cfg.Environment : HHH000206: hibernate.propertie
2019-11-20 16:36:10.491 INFO 3760 --- [main] o.hibernate.annotations.common.Version : HCANN000001: Hibernate Common
2019-11-20 16:36:11.006 INFO 3760 --- [main] org.hibernate.dialect.Dialect : HHH000400: Using dialect: org.
2019-11-20 16:36:11.319 INFO 3760 --- [main] org.hibernate.type.BasicTypeRegistry : HHH000270: Type registration [
2019-11-20 16:36:11.319 INFO 3760 --- [main] org.hibernate.type.BasicTypeRegistry : HHH000270: Type registration [
2019-11-20 16:36:11.322 INFO 3760 --- [main] org.hibernate.type.BasicTypeRegistry : HHH000270: Type registration [
2019-11-20 16:36:11.323 INFO 3760 --- [main] org.hibernate.type.BasicTypeRegistry : HHH000270: Type registration [
2019-11-20 16:36:15.511 INFO 3760 --- [main] com.scissors.auth.WebApplication : Started WebApplication in 9.94
2019-11-20 16:36:15.670 INFO 3760 --- [main] org.apache.jasper.servlet.TldScanner : At least one JAR was scanned f
2019-11-20 16:36:15.716 INFO 3760 --- [main] org.apache.coyote.http11.Http11Protocol : Starting ProtocolHandler ["ajp
2019-11-20 16:36:15.732 INFO 3760 --- [main] org.apache.catalina.startup.Catalina : Server startup in [17,814] mil
2019-11-20 16:36:26.819 INFO 3760 --- [nio-8080-exec-2] o.a.c.c.C.[.localhost].[/Scissors] : Initializing Spring Dispatcher
2019-11-20 16:36:27.175 INFO 3760 --- [nio-8080-exec-2] o.h.h.l.QueryTranslatorFactoryInitiator : HHH000397: Using ASTQueryTrans
Hibernate: select company0.COMPANY_ID as COMPANY_ID1_3_, company0.CONTACT_FNAME as CONTACT_FNAME2_3_, company0.CONTACT_LNAME as
Hibernate: {call UPD_TOT_PRESC_COMPANY()}
Hibernate: select company0.COMPANY_ID as COMPANY_ID1_3_, company0.CONTACT_FNAME as CONTACT_FNAME2_3_, company0.CONTACT_LNAME as

```



## 10. Web Application

This project was developed using the following technologies:

Computer language: Java 8

Back-end framework: Spring Boot Framework 2.0.6

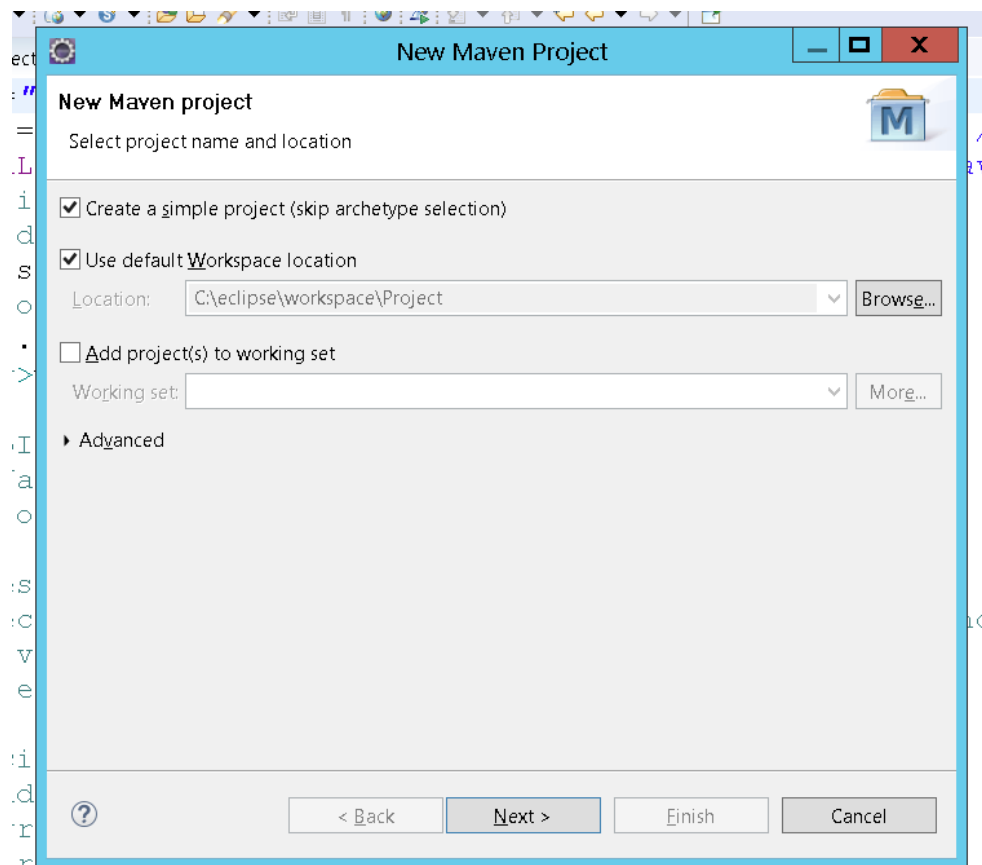
Front-end library: Bootstrap 4

Server application: Tomcat 9.0.26

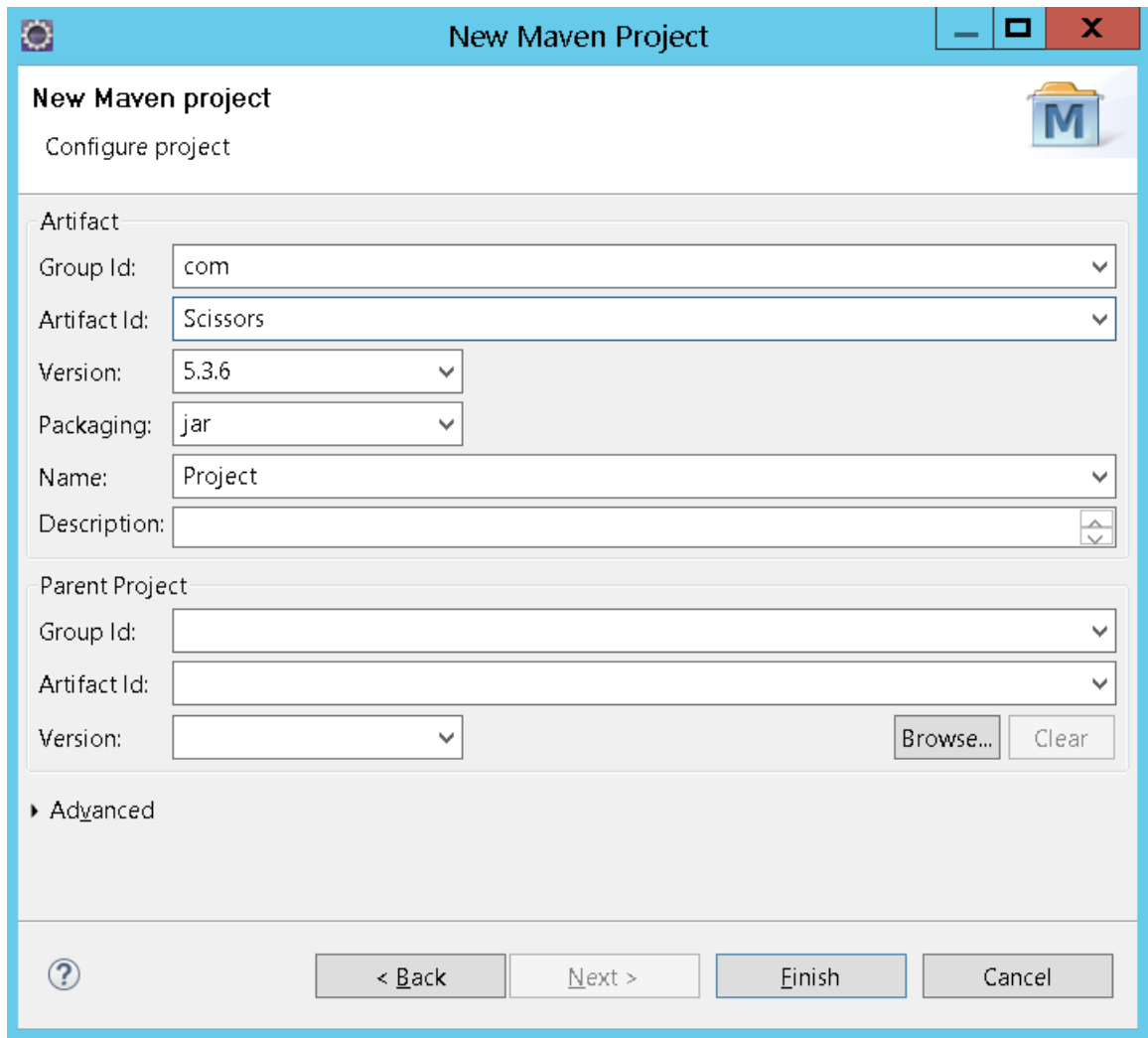
Database: Oracle 11g

It is essential to run the script to set up the database.

Create new Maven Project and click Next



Setting up the project and click Finish.



**New Maven Project**

Configure project

**Artifact**

Group Id:

Artifact Id:

Version:

Packaging:

Name:

Description:

**Parent Project**

Group Id:

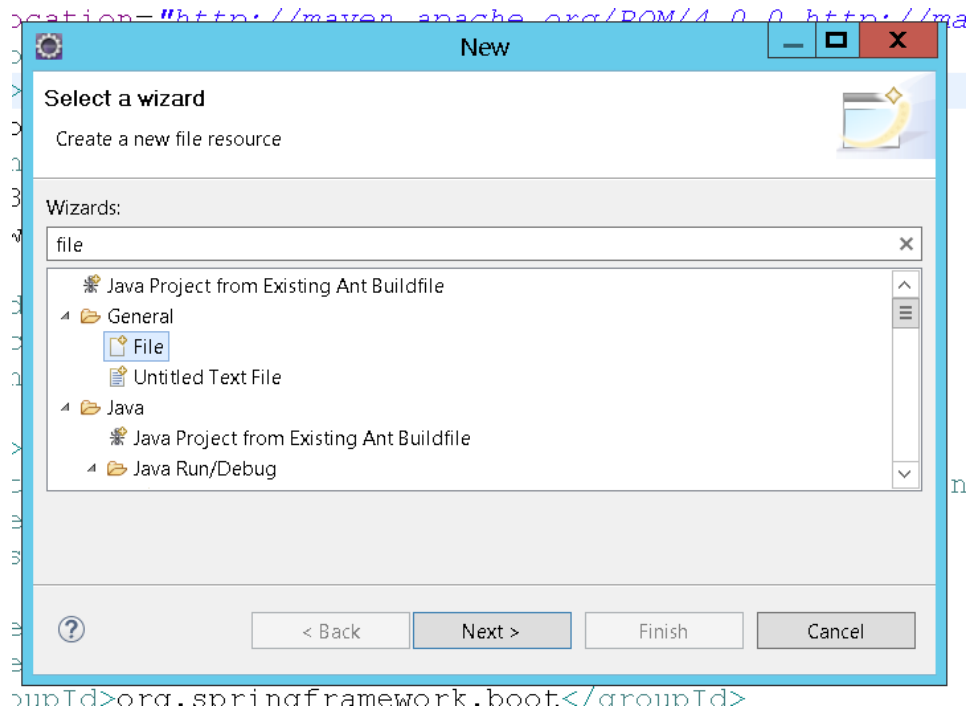
Artifact Id:

Version:

► Advanced

Setting up the application properties

Create new File and click Next



Select the folder resources, enter the file name and click Finish

**New Maven project**  
Configure project

**Artifact**

Group Id:

Artifact Id:

Version:

Packaging:

Name:

Description:

**Parent Project**

Group Id:

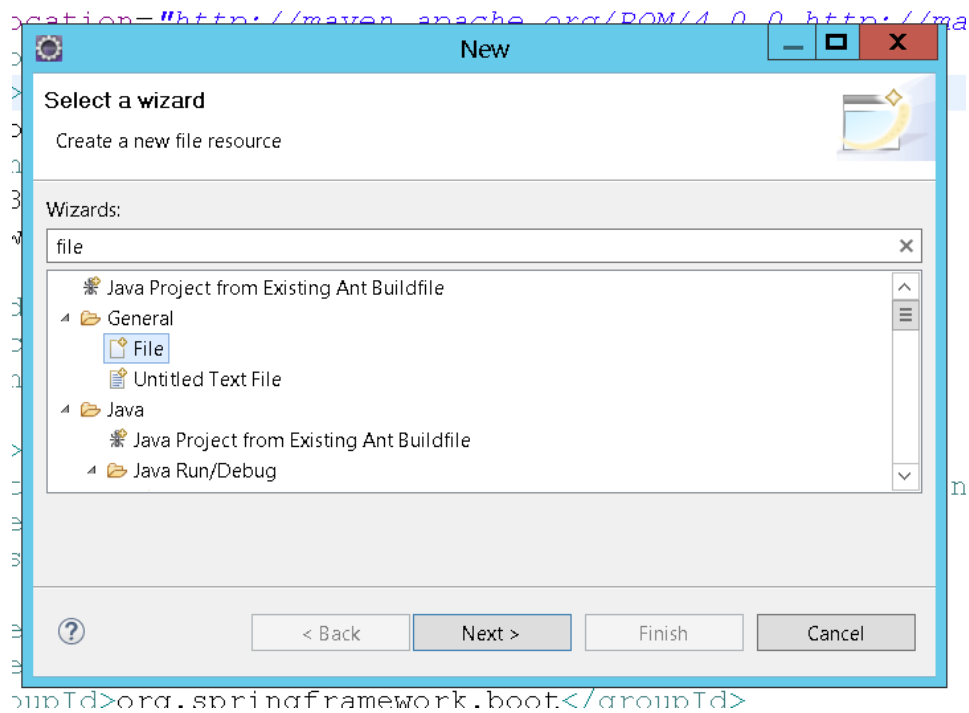
Artifact Id:

Version:

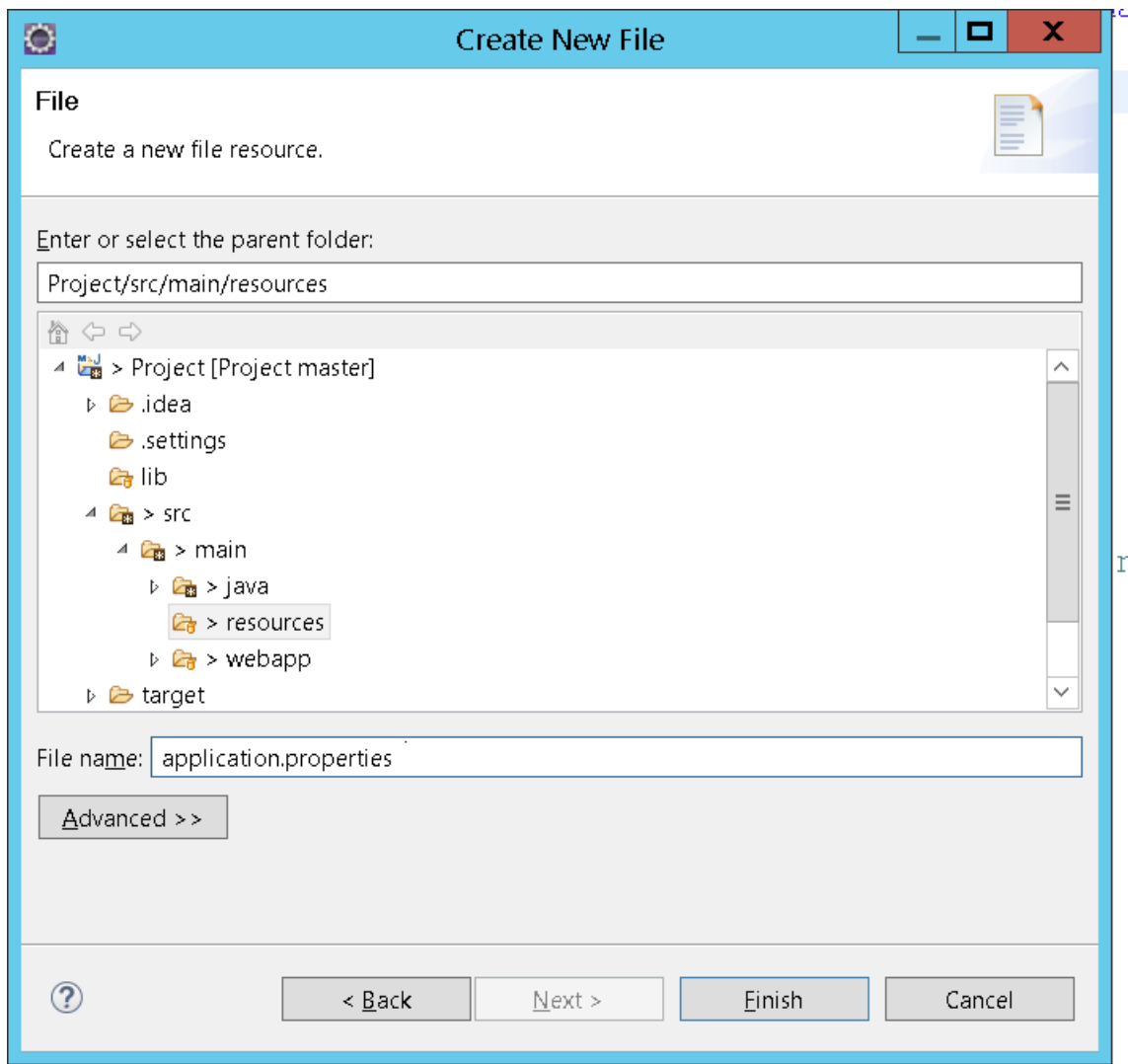
► Advanced

Setting up the application properties

Create new File and click Next



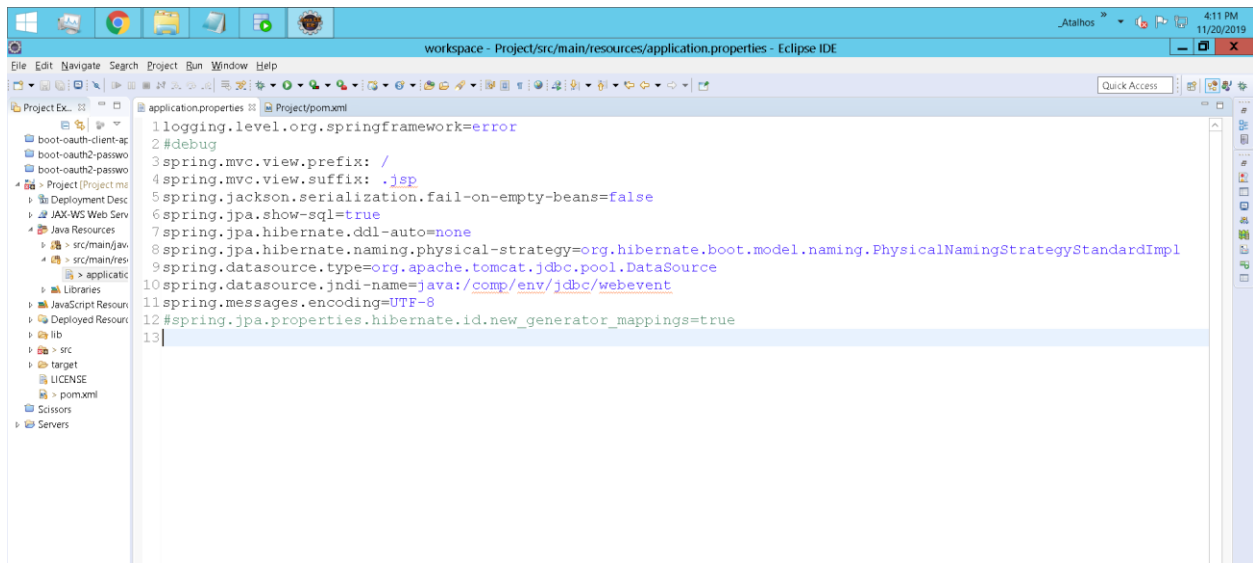
Select the folder resources, enter the file name and click Finish



artifactId>spring-boot-starter-data-jpa</artifactId>



Enter the information a below:



## Importing the libraries

Edit the POM.xml and add these dependencies:

```
<parent>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-parent</artifactId>
  <version>2.1.3.RELEASE</version>
</parent>

<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
  </dependency>

  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-jpa</artifactId>
  </dependency>

  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-jdbc</artifactId>
    <scope>provided</scope>
  </dependency>

  <dependency>
    <groupId>org.apache.tomcat.embed</groupId>
```

```

        <artifactId>tomcat-embed-jasper</artifactId>
        <scope>provided</scope>
    </dependency>

    <dependency>
        <groupId>javax.servlet</groupId>
        <artifactId>servlet-api</artifactId>
        <version>2.5</version>
        <scope>provided</scope>
    </dependency>

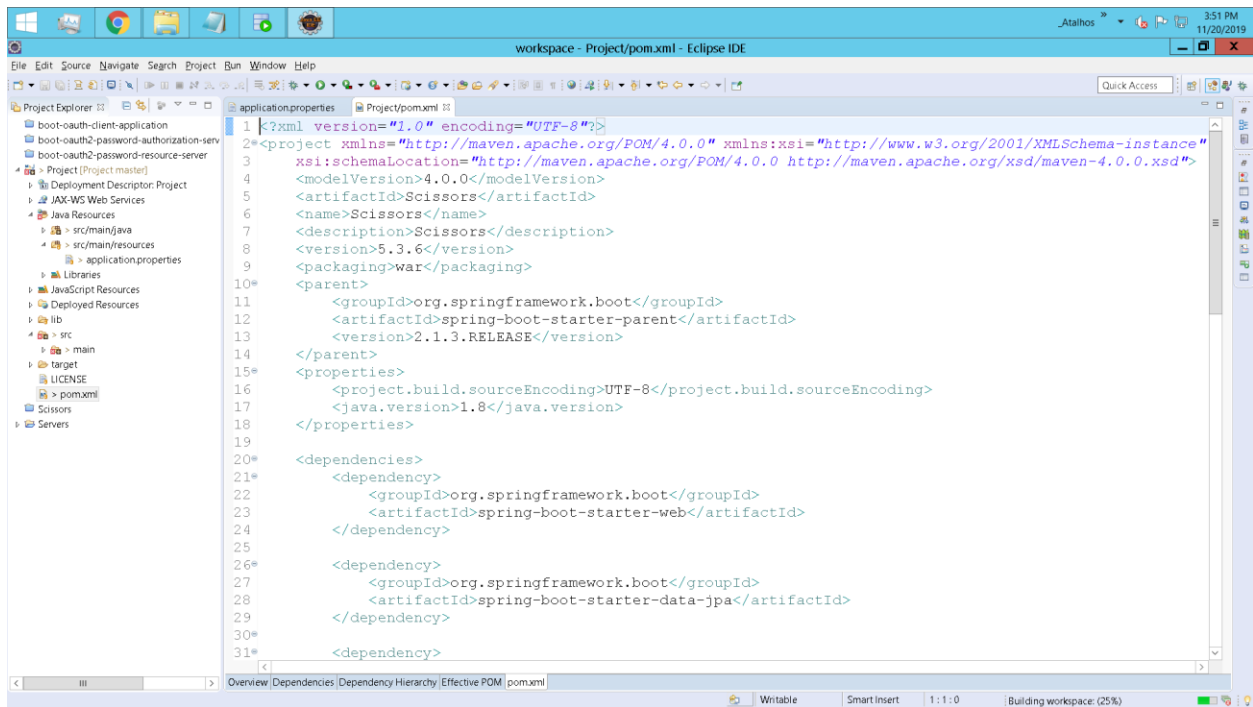
    <dependency>
        <groupId>javax</groupId>
        <artifactId>javaee-api</artifactId>
        <version>6.0</version>
        <scope>provided</scope>
    </dependency>

    <dependency>
        <groupId>javax.servlet</groupId>
        <artifactId>jstl</artifactId>
    </dependency>

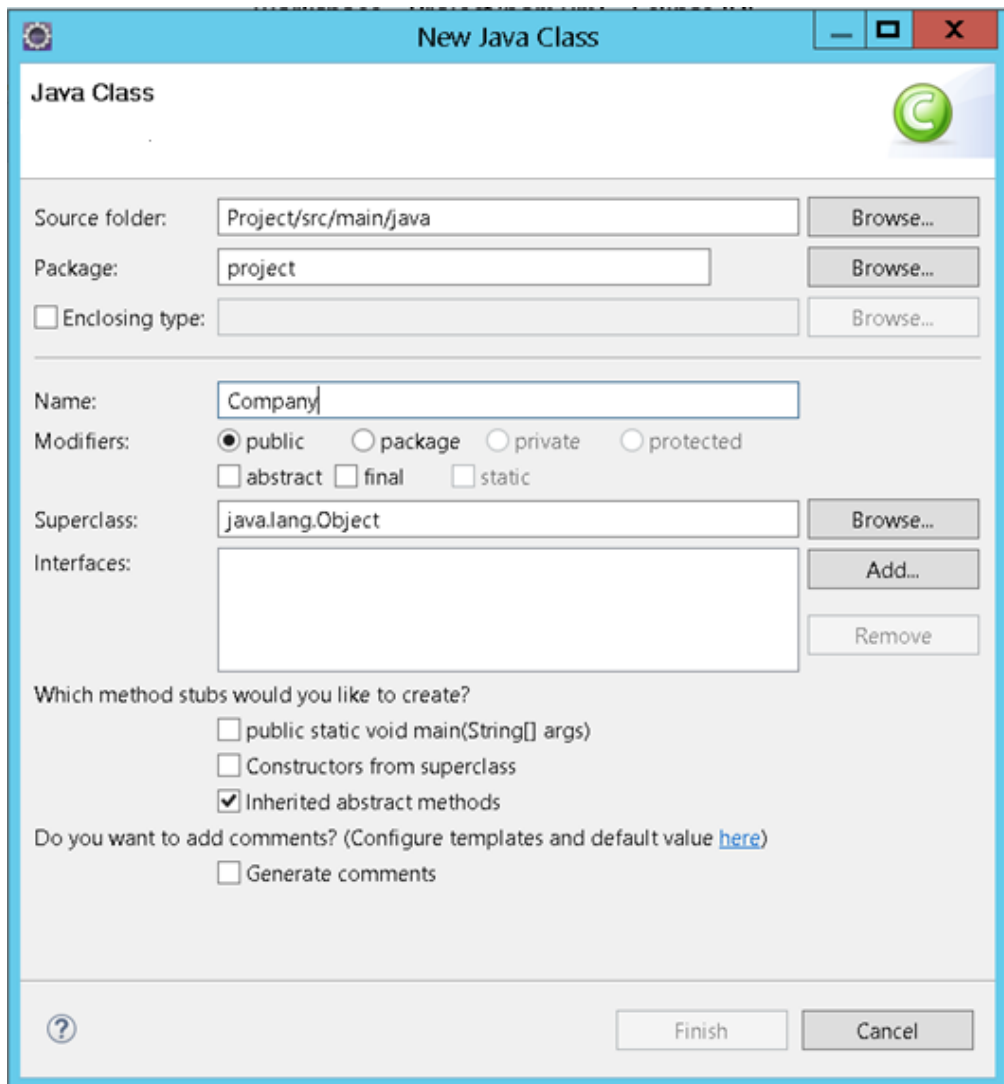
    <dependency>
        <groupId>com.oracle</groupId>
        <artifactId>ojdbc7</artifactId>
        <version>12.1.0.2</version>
        <scope>system</scope>
        <systemPath>${basedir}/lib/ojdbc7.jar</systemPath>
    </dependency>

    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-configuration-processor</artifactId>
        <optional>true</optional>
    </dependency>
</dependencies>

```



## Creating classes



The image shows a 'New Java Class' dialog box from an IDE. The title bar is blue with standard window controls. The dialog has a light gray background. At the top, it says 'Java Class' with a green circular icon. Below this, there are three rows for specifying the location: 'Source folder' with the text 'Project/src/main/java' and a 'Browse...' button; 'Package' with the text 'project' and a 'Browse...' button; and 'Enclosing type' which is currently empty with a 'Browse...' button. A horizontal line separates these from the class details. The 'Name' field contains 'Company'. The 'Modifiers' section has radio buttons for 'public' (selected), 'package', 'private', and 'protected', and checkboxes for 'abstract', 'final', and 'static'. The 'Superclass' field contains 'java.lang.Object' with a 'Browse...' button. The 'Interfaces' section has an empty text area, an 'Add...' button, and a 'Remove' button. Below this, a section titled 'Which method stubs would you like to create?' has three checkboxes: 'public static void main(String[] args)' (unchecked), 'Constructors from superclass' (unchecked), and 'Inherited abstract methods' (checked). Another section titled 'Do you want to add comments? (Configure templates and default value [here](#))' has a 'Generate comments' checkbox (unchecked). At the bottom, there is a help icon (question mark in a circle) on the left, and 'Finish' and 'Cancel' buttons on the right.

**New Java Class**

Java Class

Source folder:

Package:

☐ Enclosing type:

---

Name:

Modifiers: ☒ public ☐ package ☐ private ☐ protected  
☐ abstract ☐ final ☐ static

Superclass:

Interfaces:

Which method stubs would you like to create?

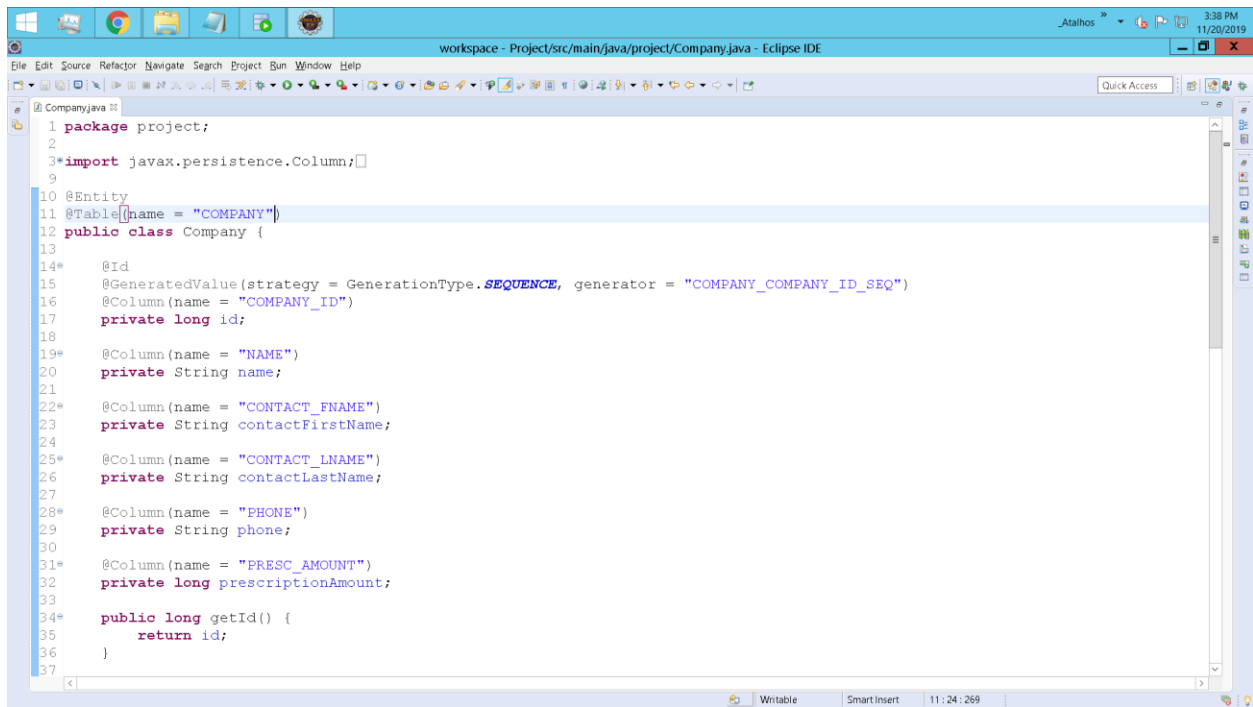
☐ public static void main(String[] args)

☐ Constructors from superclass

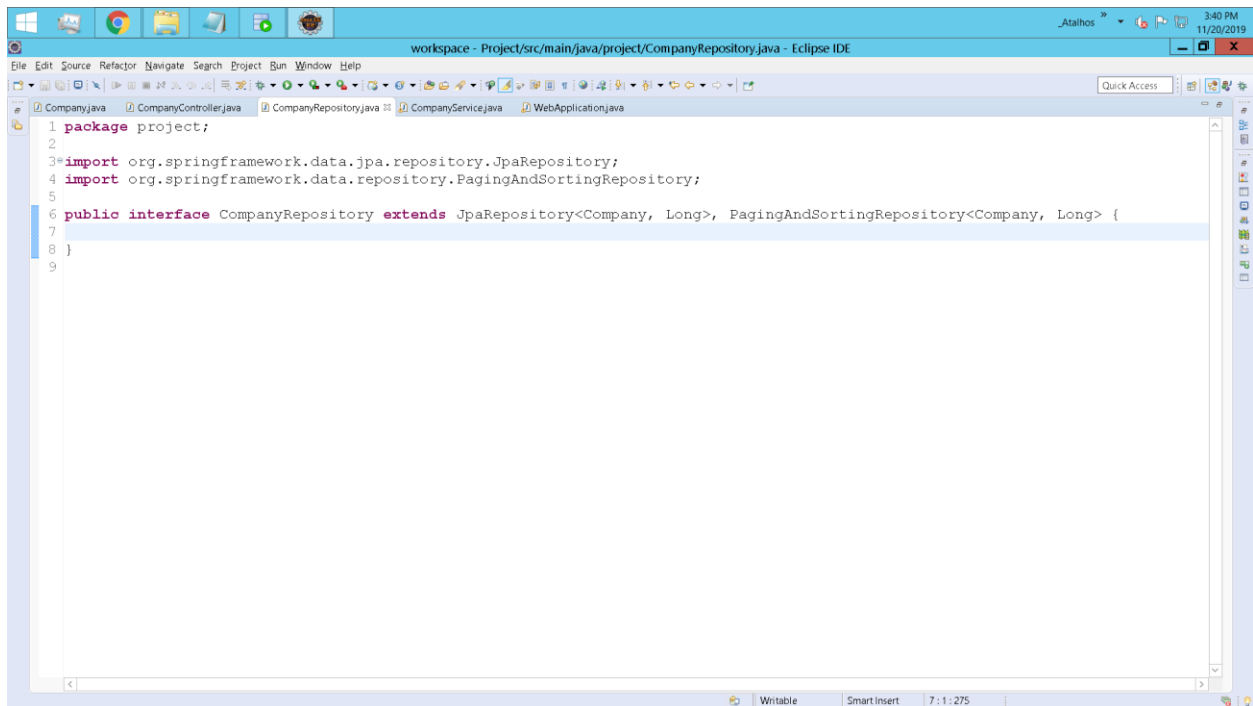
☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))

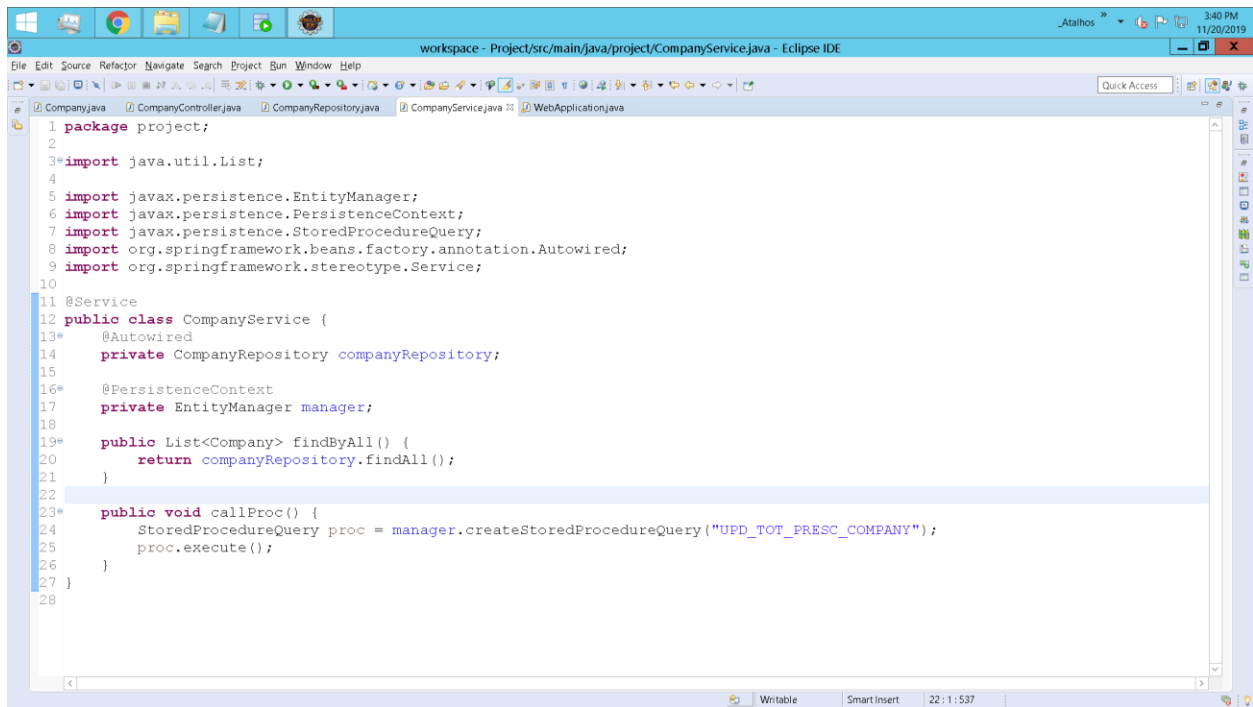
☐ Generate comments



```
1 package project;
2
3 import javax.persistence.Column;
4
5
6
7
8
9
10 @Entity
11 @Table(name = "COMPANY")
12 public class Company {
13
14     @Id
15     @GeneratedValue(strategy = GenerationType.SEQUENCE, generator = "COMPANY_COMPANY_ID_SEQ")
16     @Column(name = "COMPANY_ID")
17     private long id;
18
19     @Column(name = "NAME")
20     private String name;
21
22     @Column(name = "CONTACT_FNAME")
23     private String contactFirstName;
24
25     @Column(name = "CONTACT_LNAME")
26     private String contactLastName;
27
28     @Column(name = "PHONE")
29     private String phone;
30
31     @Column(name = "PRESC_AMOUNT")
32     private long prescriptionAmount;
33
34     public long getId() {
35         return id;
36     }
37 }
```



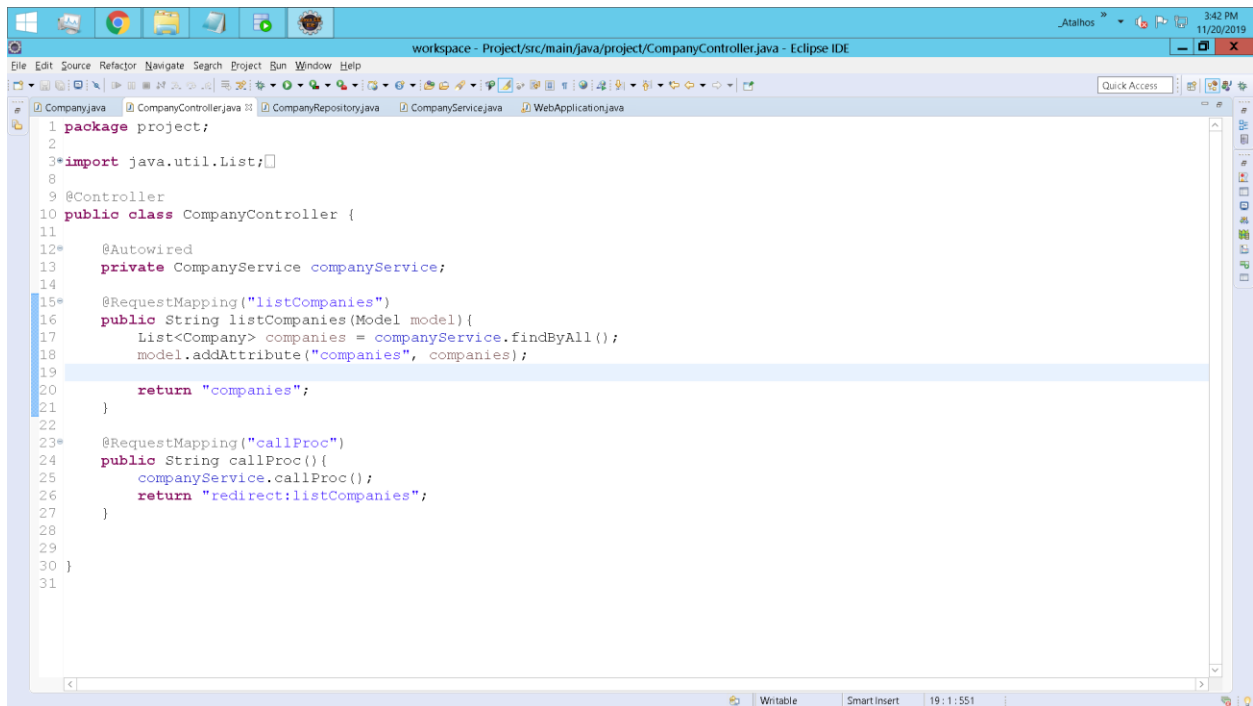
```
1 package project;
2
3 import org.springframework.data.jpa.repository.JpaRepository;
4 import org.springframework.data.repository.PagingAndSortingRepository;
5
6 public interface CompanyRepository extends JpaRepository<Company, Long>, PagingAndSortingRepository<Company, Long> {
7
8 }
9 }
```



The screenshot shows the Eclipse IDE with the file `CompanyService.java` open. The code is as follows:

```
1 package project;
2
3 import java.util.List;
4
5 import javax.persistence.EntityManager;
6 import javax.persistence.PersistenceContext;
7 import javax.persistence.StoredProcedureQuery;
8 import org.springframework.beans.factory.annotation.Autowired;
9 import org.springframework.stereotype.Service;
10
11 @Service
12 public class CompanyService {
13     @Autowired
14     private CompanyRepository companyRepository;
15
16     @PersistenceContext
17     private EntityManager manager;
18
19     public List<Company> findByAll() {
20         return companyRepository.findAll();
21     }
22
23     public void callProc() {
24         StoredProcedureQuery proc = manager.createStoredProcedureQuery("UPD_TOT_PRESC_COMPANY");
25         proc.execute();
26     }
27 }
28
```

The IDE interface includes a menu bar (File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help), a toolbar, and a package explorer on the left showing the project structure: `Company.java`, `CompanyController.java`, `CompanyRepository.java`, `CompanyService.java`, and `WebApplication.java`. The status bar at the bottom indicates 'Writable', 'Smart Insert', and the time '22:1:537'.



The screenshot shows the Eclipse IDE with the file `CompanyController.java` open. The code is as follows:

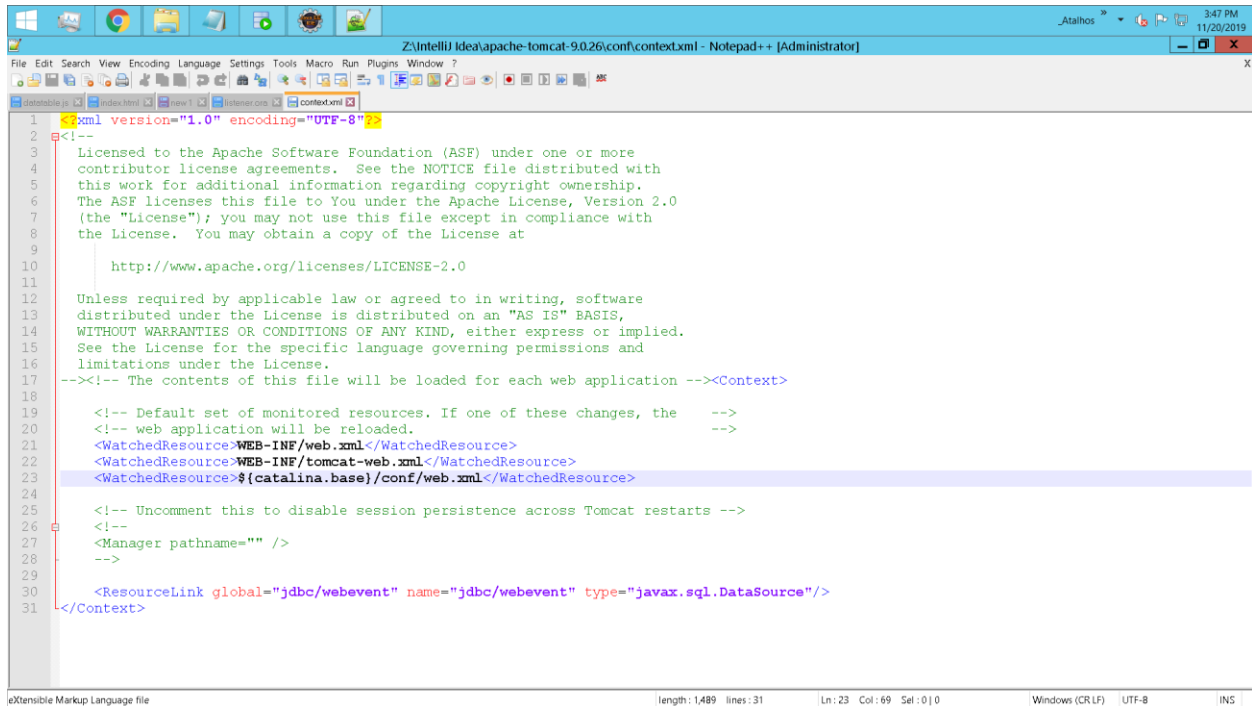
```
1 package project;
2
3 import java.util.List;
4
5 @Controller
6 public class CompanyController {
7
8     @Autowired
9     private CompanyService companyService;
10
11     @RequestMapping("listCompanies")
12     public String listCompanies(Model model) {
13         List<Company> companies = companyService.findByAll();
14         model.addAttribute("companies", companies);
15
16         return "companies";
17     }
18
19     @RequestMapping("callProc")
20     public String callProc() {
21         companyService.callProc();
22         return "redirect:listCompanies";
23     }
24 }
25
26
27
28
29
30
31
```

The IDE interface is similar to the first screenshot, with the package explorer showing the same project structure. The status bar at the bottom indicates 'Writable', 'Smart Insert', and the time '19:1:551'.



Download Tomcat : <http://us.mirrors.quenda.co/apache/tomcat/tomcat-9/v9.0.26/bin/apache-tomcat-9.0.26-windows-x86.zip>

Add the resource link tag in the content.xml located in tomcat\conf\context.xml

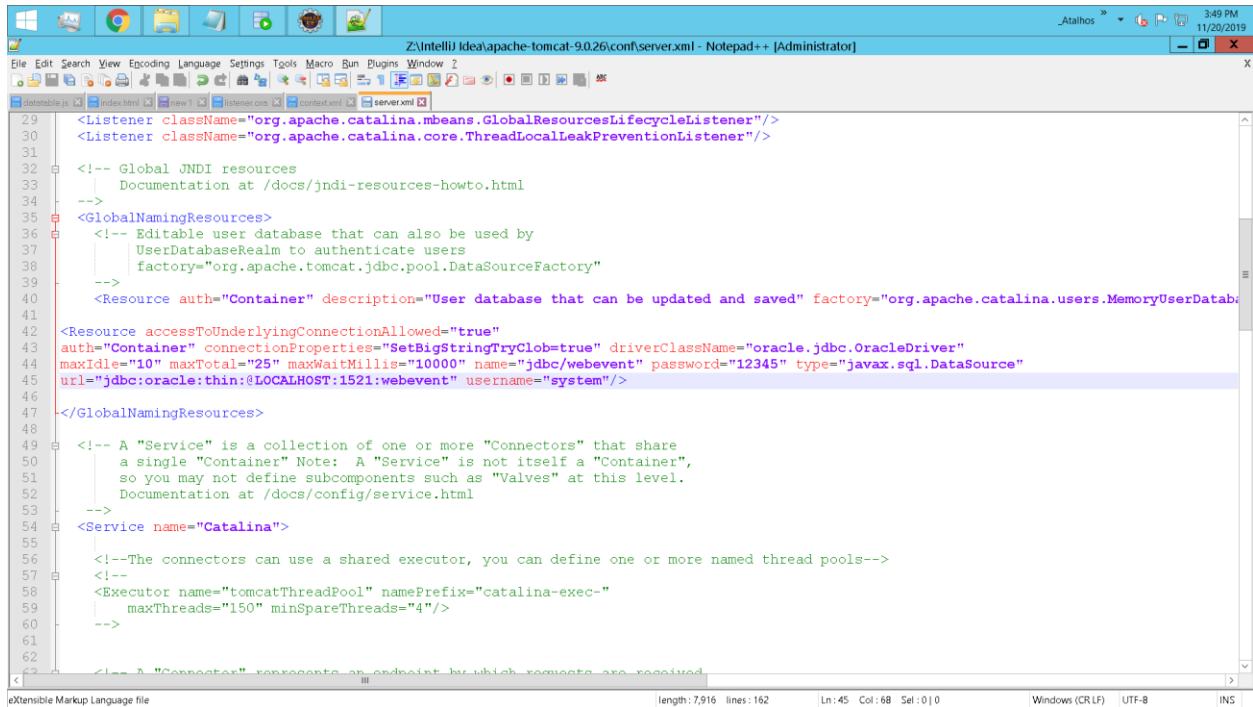


The screenshot shows a Notepad++ window titled "Z:\IntelliJ Idea\apache-tomcat-9.0.26\conf\context.xml - Notepad++ [Administrator]". The file content is as follows:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!--
3 Licensed to the Apache Software Foundation (ASF) under one or more
4 contributor license agreements. See the NOTICE file distributed with
5 this work for additional information regarding copyright ownership.
6 The ASF licenses this file to You under the Apache License, Version 2.0
7 (the "License"); you may not use this file except in compliance with
8 the License. You may obtain a copy of the License at
9
10 http://www.apache.org/licenses/LICENSE-2.0
11
12 Unless required by applicable law or agreed to in writing, software
13 distributed under the License is distributed on an "AS IS" BASIS,
14 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
15 See the License for the specific language governing permissions and
16 limitations under the License.
17 --><!-- The contents of this file will be loaded for each web application --><Context>
18
19 <!-- Default set of monitored resources. If one of these changes, the -->
20 <!-- web application will be reloaded. -->
21 <WatchedResource>WEB-INF/web.xml</WatchedResource>
22 <WatchedResource>WEB-INF/tomcat-web.xml</WatchedResource>
23 <WatchedResource>${catalina.base}/conf/web.xml</WatchedResource>
24
25 <!-- Uncomment this to disable session persistence across Tomcat restarts -->
26 <!--
27 <Manager pathname="" />
28 -->
29
30 <ResourceLink global="jdbc/webevent" name="jdbc/webevent" type="javax.sql.DataSource"/>
31 </Context>
```

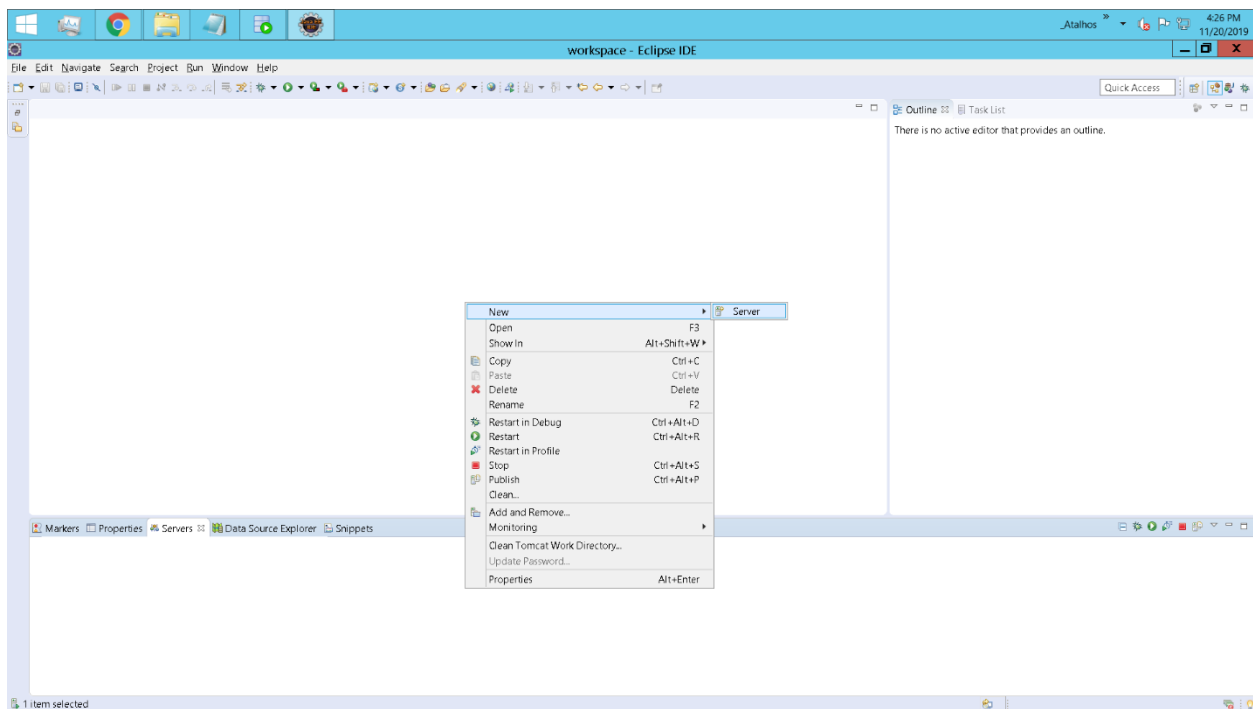
The status bar at the bottom indicates: "eXtensible Markup Language file", "length: 1,489 lines: 31", "Ln: 23 Col: 69 Sel: 0 | 0", "Windows (CRLF)", "UTF-8", and "INS".

Add the resource tag in the server.xml located in tomcat\conf\server.xml



```
29 <Listener className="org.apache.catalina.mbeans.GlobalResourcesLifecycleListener"/>
30 <Listener className="org.apache.catalina.core.ThreadLocalLeakPreventionListener"/>
31
32 <!-- Global JNDI resources
33      Documentation at /docs/jndi-resources-howto.html
34 -->
35 <GlobalNamingResources>
36     <!-- Editable user database that can also be used by
37          UserDatabaseRealm to authenticate users
38          factory="org.apache.tomcat.jdbc.pool.DataSourceFactory"
39     -->
40     <Resource auth="Container" description="User database that can be updated and saved" factory="org.apache.catalina.users.MemoryUserDatabase"
41     <Resource accessToUnderlyingConnectionAllowed="true"
42     auth="Container" connectionProperties="SetBigStringTryClob=true" driverClassName="oracle.jdbc.OracleDriver"
43     maxIdle="10" maxTotal="25" maxWaitMillis="10000" name="jdbc/webevent" password="12345" type="javax.sql.DataSource"
44     url="jdbc:oracle:thin:@LOCALHOST:1521:webevent" username="system"/>
45 </GlobalNamingResources>
46
47 <!-- A "Service" is a collection of one or more "Connectors" that share
48      a single "Container" Note: A "Service" is not itself a "Container",
49      so you may not define subcomponents such as "Valves" at this level.
50      Documentation at /docs/config/service.html
51 -->
52 <Service name="Catalina">
53     <!--The connectors can use a shared executor, you can define one or more named thread pools-->
54     <!--
55     <Executor name="tomcatThreadPool" namePrefix="catalina-exec-"
56     maxThreads="150" minSpareThreads="4"/>
57     -->
58
59     <!-- A "Connector" represents an endpoint by which requests are received
60     -->
61
62     <!-- A "Connector" represents an endpoint by which requests are received
```

Add the tomcat server in your eclipse



New Server

Define a New Server

Choose the type of server to create

Select the server type:

type filter text

Tomcat v5.0 Server

Tomcat v5.5 Server

Tomcat v6.0 Server

Tomcat v7.0 Server

Tomcat v8.0 Server

Tomcat v8.5 Server

Tomcat v9.0 Server

Publishes and runs J2EE and Java EE Web projects and server configurations to a local Tomcat server.

Server's host name:

localhost

Server name:

Tomcat v9.0 Server at localhost

Server runtime environment:

Apache Tomcat v9.0

[Add...](#)

[Configure runtime environments...](#)

?

< Back

Next >

Finish

Cancel

Click in add to add the tomcat directory and click Finish and Finish

**New Server**

**New Server Runtime Environment**

**Tomcat Server**  
Specify the installation directory

Name:  
Apache Tomcat v9.0

Tomcat installation directory:  
Z:\IntelliJ Idea\apache-tomcat-9.0.26 [Browse...](#)  
[Download and Install...](#)

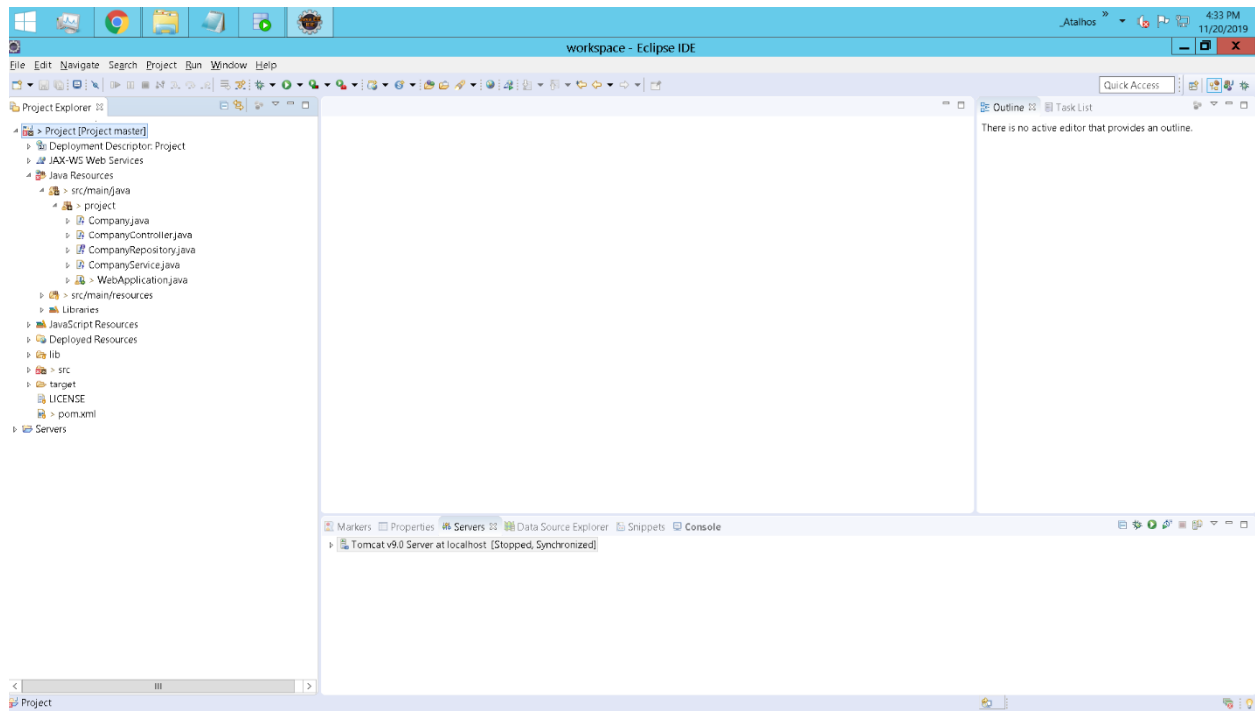
JRE:  
Workbench default JRE [Installed JREs...](#)

[?](#) [< Back](#) [Next >](#) [Finish](#) [Cancel](#)

Server runtime environment: Apache Tomcat v9.0 [Add...](#)  
[Configure runtime environments...](#)

[?](#) [< Back](#) [Next >](#) [Finish](#) [Cancel](#)

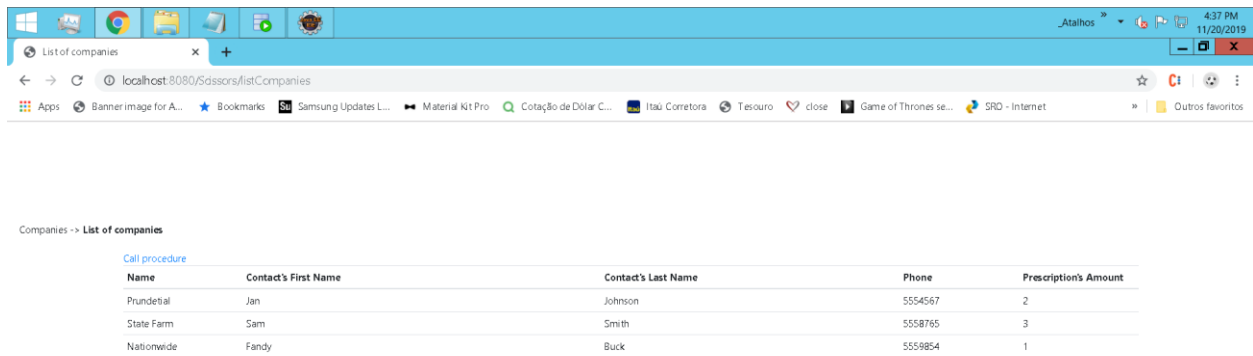
Select the project and click Run.



Companies -> List of companies

Call procedure				
Name	Contact's First Name	Contact's Last Name	Phone	Prescription's Amount
Prundetal	Jan	Johnson	5554567	0
State Farm	Sam	Smith	5558765	0
Nationwide	Fandy	Buck	5559854	0

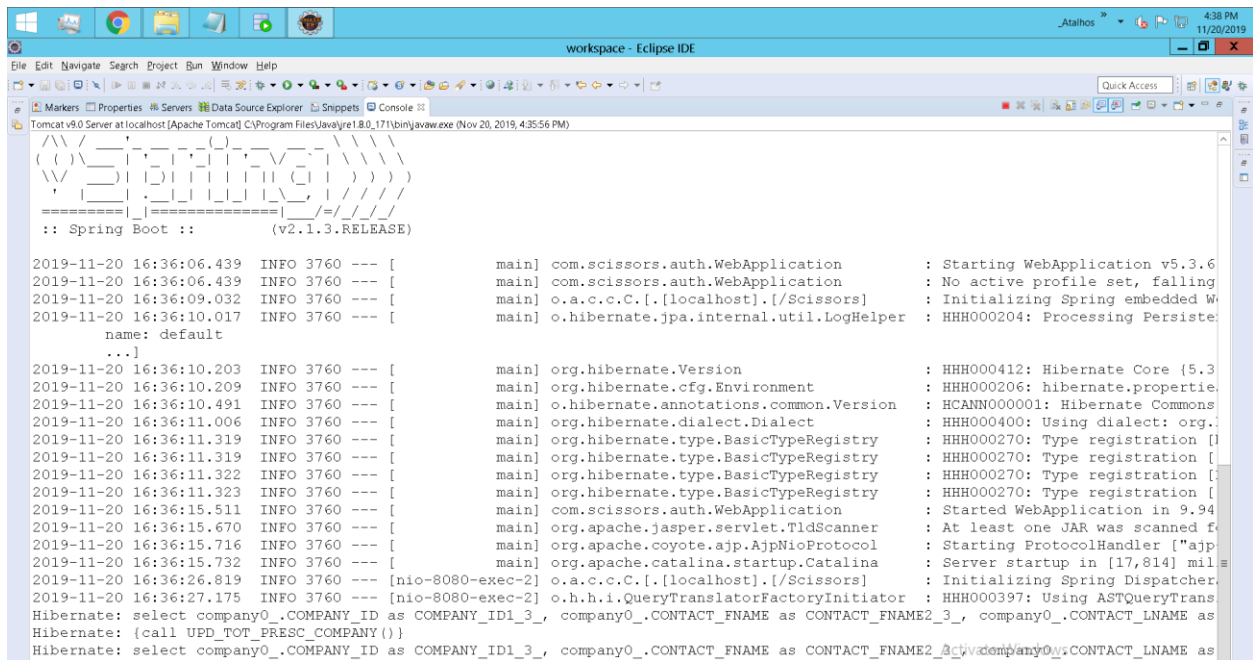
## Click in call procedure



The screenshot shows a web browser window with the address bar displaying 'localhost:8080/Scissors/listCompanies'. The page content includes a link 'Call procedure' and a table with the following data:

Name	Contact's First Name	Contact's Last Name	Phone	Prescription's Amount
Prudental	Jan	Johnson	5554567	2
State Farm	Sam	Smith	5558765	3
Nationwide	Fandy	Buck	5559854	1

## The button called the procedure



The screenshot shows the Eclipse IDE console with the following output:

```
Tomcat v9.0 Server at localhost [Apache Tomcat/9.0.17] (Nov 20, 2019, 4:35:56 PM)

:: Spring Boot :: (v2.1.3.RELEASE)

2019-11-20 16:36:06.439 INFO 3760 --- [main] com.scissors.auth.WebApplication : Starting WebApplication v5.3.6
2019-11-20 16:36:06.439 INFO 3760 --- [main] com.scissors.auth.WebApplication : No active profile set, falling
2019-11-20 16:36:09.032 INFO 3760 --- [main] o.a.c.c.C.[.localhost].[/Scissors] : Initializing Spring embedded W
2019-11-20 16:36:10.017 INFO 3760 --- [main] o.hibernate.jpa.internal.util.LogHelper : HHH000204: Processing Persiste
name: default
...
2019-11-20 16:36:10.203 INFO 3760 --- [main] org.hibernate.Version : HHH000412: Hibernate Core {5.3
2019-11-20 16:36:10.209 INFO 3760 --- [main] org.hibernate.cfg.Environment : HHH000206: hibernate.propertie
2019-11-20 16:36:10.491 INFO 3760 --- [main] o.hibernate.annotations.common.Version : HCANN000001: Hibernate Commons
2019-11-20 16:36:11.006 INFO 3760 --- [main] org.hibernate.dialect.Dialect : HHH000400: Using dialect: org.
2019-11-20 16:36:11.319 INFO 3760 --- [main] org.hibernate.type.BasicTypeRegistry : HHH000270: Type registration [
2019-11-20 16:36:11.319 INFO 3760 --- [main] org.hibernate.type.BasicTypeRegistry : HHH000270: Type registration [
2019-11-20 16:36:11.322 INFO 3760 --- [main] org.hibernate.type.BasicTypeRegistry : HHH000270: Type registration [
2019-11-20 16:36:11.323 INFO 3760 --- [main] org.hibernate.type.BasicTypeRegistry : HHH000270: Type registration [
2019-11-20 16:36:15.511 INFO 3760 --- [main] com.scissors.auth.WebApplication : Started WebApplication in 9.94
2019-11-20 16:36:15.670 INFO 3760 --- [main] org.apache.jasper.servlet.TldScanner : At least one JAR was scanned f
2019-11-20 16:36:15.716 INFO 3760 --- [main] org.apache.coyote.apache.AjpNioProtocol : Starting ProtocolHandler ["ajp
2019-11-20 16:36:15.732 INFO 3760 --- [main] org.apache.catalina.startup.Catalina : Server startup in [17,814] mil
2019-11-20 16:36:26.819 INFO 3760 --- [nio-8080-exec-2] o.a.c.c.C.[.localhost].[/Scissors] : Initializing Spring Dispatcher
2019-11-20 16:36:27.175 INFO 3760 --- [nio-8080-exec-2] o.h.h.i.QueryTranslatorFactoryInitiator : HHH000397: Using ASTQueryTrans
Hibernate: select company0_.COMPANY_ID as COMPANY_ID1_3_, company0_.CONTACT_FNAME as CONTACT_FNAME2_3_, company0_.CONTACT_LNAME as
Hibernate: {call UPD_TOT_PRESC_COMPANY()}
```