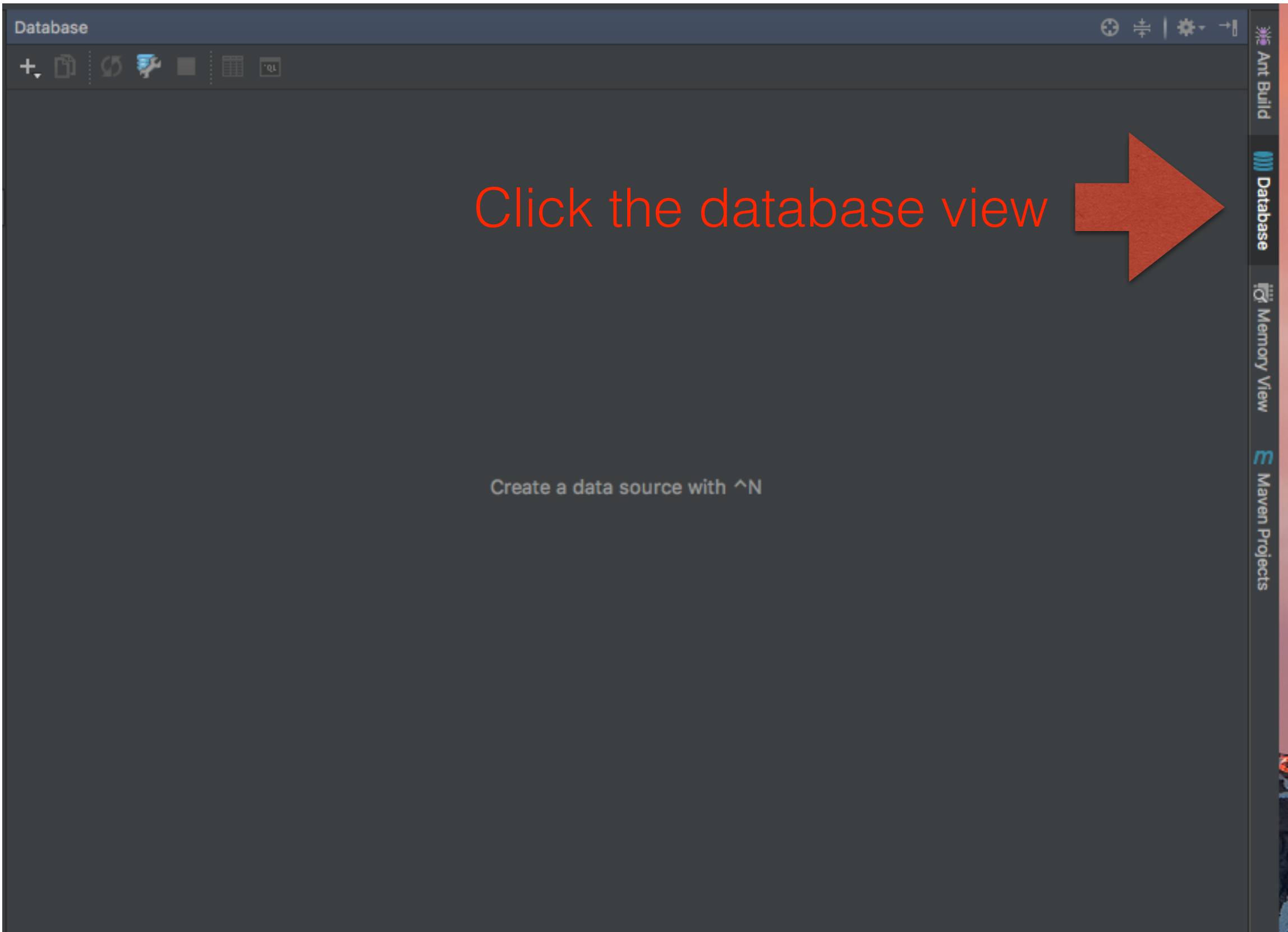
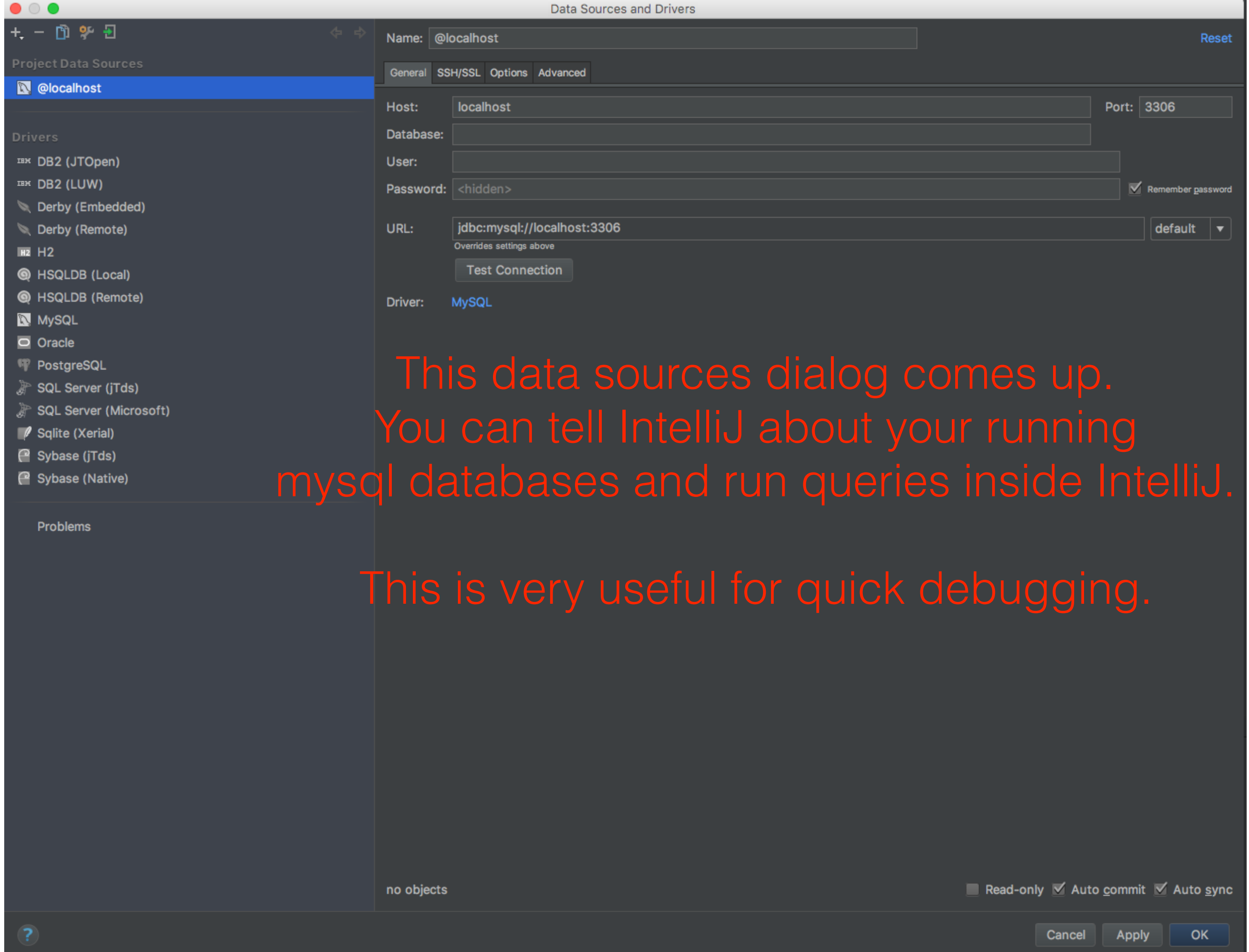


Setting up a Database Source in IntelliJ

Using SimpleAffableBean



Click the database view



This data sources dialog comes up.
You can tell IntelliJ about your running
mysql databases and run queries inside IntelliJ.

This is very useful for quick debugging.

Data Sources and Drivers

Name: MySQL

Settings Advanced

Class: com.mysql.jdbc.Driver

Dialect: MySQL

☒ Auto commit ☒ Auto sync

JDBC drivers

☒ Use provided driver [MySQL Connector/J \[latest\]](#)

Additional:

[/Users/satkinson/Library/Preferences/IntelliJ2016.3/jdbc-drivers/jdbc-drivers.xml](#)

Project Data Sources

@localhost

Drivers

- DB2 (JTOpen)
- DB2 (LUW)
- Derby (Embedded)
- Derby (Remote)
- H2
- MySQL
- Oracle
- PostgreSQL
- SQL Server (jTds)
- SQL Server (Microsoft)
- SQLite (Xerial)
- Sybase (jTds)
- Sybase (Native)

Problems

URL templates

Name	Template
default	jdbc:mysql://{host::localhost}?[{port::3306}][/{database}?][\?&,user={user},password={password},{:identifier}={:identifier...}
default	jdbc:mysql://address=\\(protocol=tcp\\)(\\(<\\)(,host={host:host_ipv6:localhost},port={port::3306},user={user},password={pa...

Cancel Apply OK

You can double-check which driver IntelliJ is using, by choosing the MySQL driver.

I just use the internally chosen driver here since we are really just debugging using IntelliJ.

Nothing needs to change.

Data Sources and Drivers

Name: [Reset](#)

General SSH/SSL Options Advanced

Host: Port:

Database:

User:

Password: ☒ Remember password

URL: default

Overrides settings above

Driver: **MySQL**

I'm using "afb1" as username, database name and password.

Feel free to use anything you have set up in your database installation.

no objects ☐ Read-only ☒ Auto commit ☒ Auto sync

Error: [08S01] Communications link failure ([view](#))

Project Data Sources

afb1@localhost

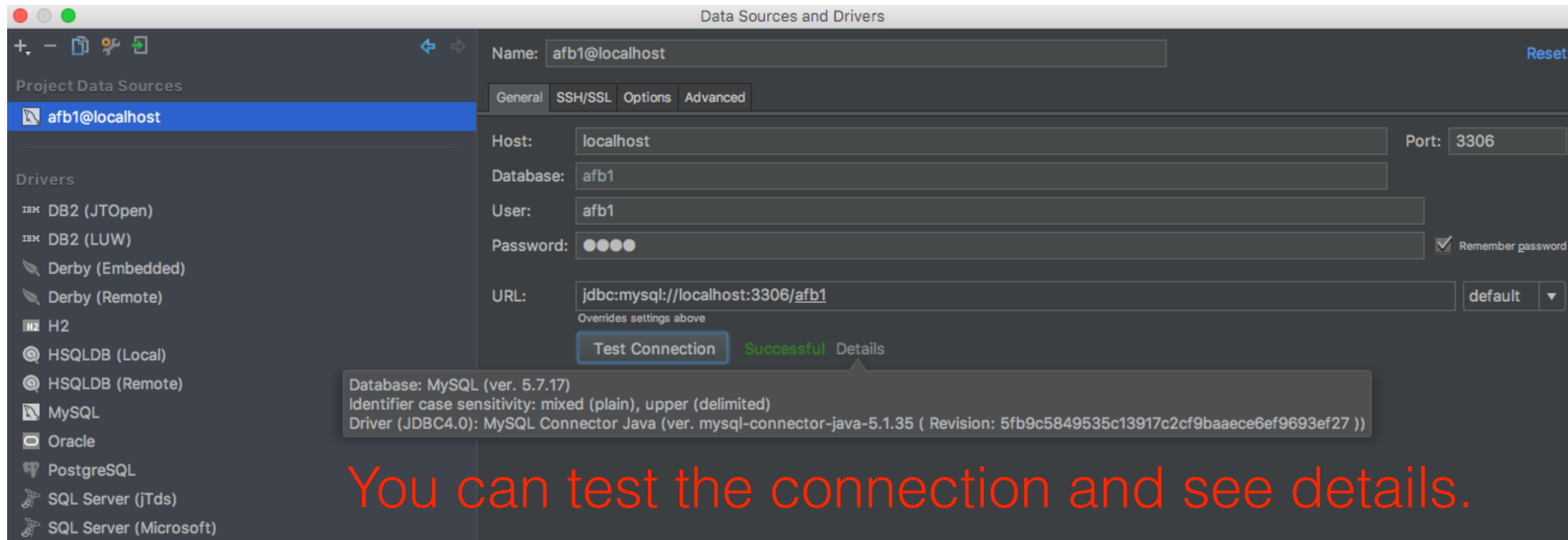
Drivers

- IBM DB2 (JTOpen)
- IBM DB2 (LUW)
- Derby (Embedded)
- Derby (Remote)
- H2
- HSQLDB (Local)
- HSQLDB (Remote)
- MySQL
- Oracle
- PostgreSQL
- SQL Server (jTds)
- SQL Server (Microsoft)
- SQLite (Xerial)
- Sybase (jTds)
- Sybase (Native)

Problems

```
$ mysql.server start
```

Don't forget to run the database!



You can test the connection and see details.

Now you can execute the schemaCreation.sql included in the project right from IntelliJ!

Browse to the SQL

Right click and choose "Run <filename>" to run the whole file at once.

Do this for the schema creation, then for the sample data.

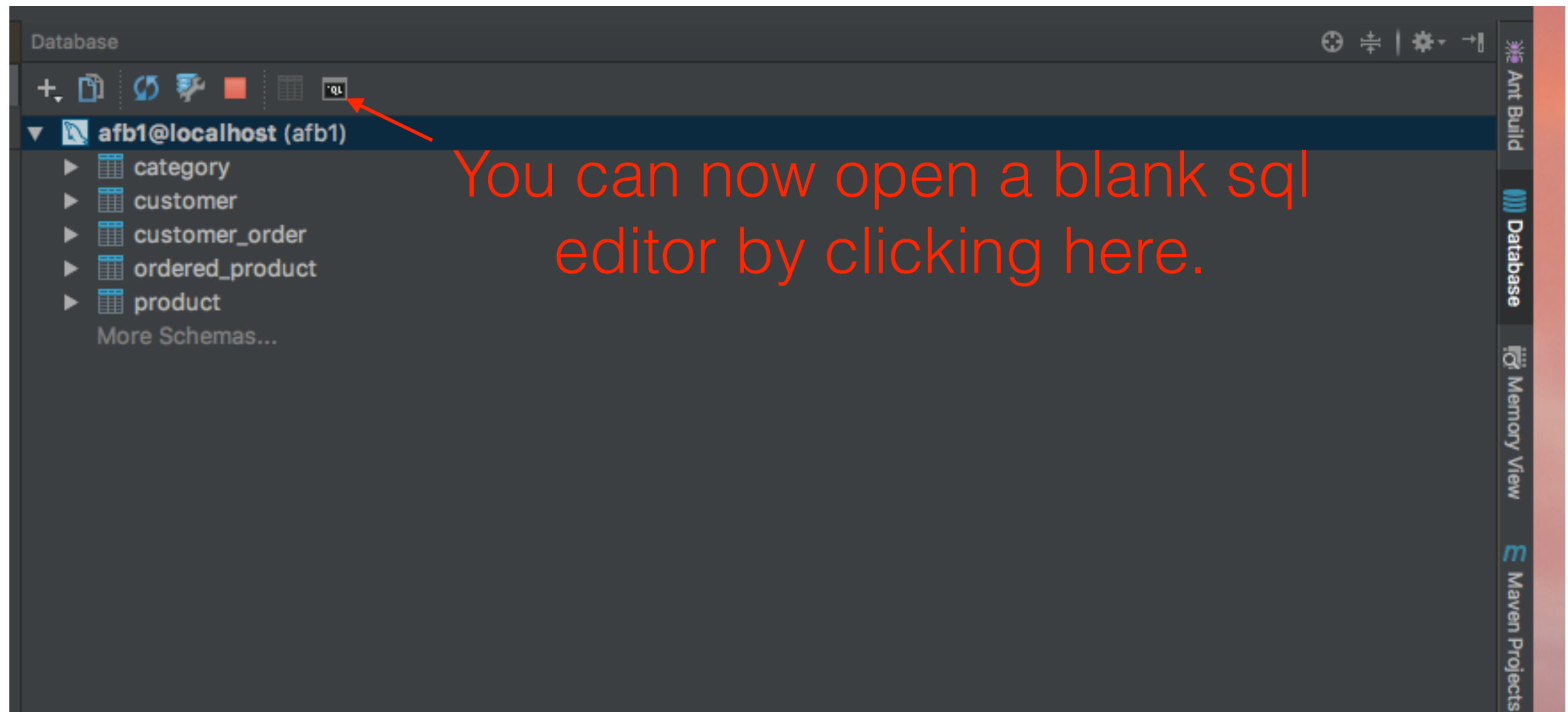
```
-- Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
--
-- You may not modify, use, reproduce, or distribute this software
-- except in compliance with the terms of the license at:
-- http://developer.sun.com/berkeley_license.html
--
-- author: tgiunipero
--
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='TRADITIONAL';

DROP SCHEMA IF EXISTS `affablebean` ;
CREATE SCHEMA IF NOT EXISTS `affablebean` DEFAULT CHARACTER SET utf8 COLLATE utf8_unicode_ci ;
USE `affablebean` ;

-- Table `affablebean`.`customer`
--
DROP TABLE IF EXISTS `affablebean`.`customer` ;

CREATE TABLE IF NOT EXISTS `affablebean`.`customer` (
  `id` INT UNSIGNED NOT NULL AUTO_INCREMENT ,
  `name` VARCHAR(45) NOT NULL ,
  `email` VARCHAR(45) NOT NULL ,
  `phone` VARCHAR(45) NOT NULL ,
  `address` VARCHAR(45) NOT NULL ,
  `city_region` VARCHAR(2) NOT NULL ,
  `cc_number` VARCHAR(19) NOT NULL ,
  PRIMARY KEY (`id` ) )
ENGINE = InnoDB
COMMENT = 'maintains customer details';
```

Your SQL schema (tables) should now show up in the database view.



You can run a query (or update) and see results in IntelliJ below in the IDE itself. Very useful.

The screenshot shows the IntelliJ IDEA IDE interface. The top toolbar includes the 'Run' button (a green play icon). The main editor window displays a SQL query: `select * from product`. The left sidebar shows the project structure, including folders like `admin`, `css`, `img`, and `products`. The bottom panel shows the 'Database Console' with a table of results for the query. The table has columns: `id`, `name`, `price`, `description`, `last_update`, and `category_id`. The results show 16 rows of product data.

id	name	price	description	last_update	category_id
1	milk	1.70	semi skimmed (1L)	2017-01-07 12:55:18	1
2	cheese	2.39	mild cheddar (330g)	2017-01-07 12:55:18	1
3	butter	1.09	unsalted (250g)	2017-01-07 12:55:18	1
4	free range eggs	1.76	medium-sized (6 eggs)	2017-01-07 12:55:18	1
5	organic meat patties	2.29	rolled in fresh herbs 2 patties (250g)	2017-01-07 12:55:18	2
6	parma ham	3.49	matured, organic (70g)	2017-01-07 12:55:18	2
7	chicken leg	2.59	free range (250g)	2017-01-07 12:55:18	2
8	sausages	3.55	reduced fat, pork 3 sausages (350g)	2017-01-07 12:55:18	2
9	sunflower seed loaf	1.89	600g	2017-01-07 12:55:18	3
10	sesame seed bagel	1.19	4 bagels	2017-01-07 12:55:18	3
11	pumpkin seed bun	1.15	4 buns	2017-01-07 12:55:18	3
12	chocolate cookies	2.39	contain peanuts (3 cookies)	2017-01-07 12:55:18	3
13	corn on the cob	1.59	2 pieces	2017-01-07 12:55:18	4
14	red currants	2.49	150g	2017-01-07 12:55:18	4
15	broccoli	1.29	500g	2017-01-07 12:55:18	4
16	seedless watermelon	1.49	750g	2017-01-07 12:55:18	4

Happy SQL hacking!