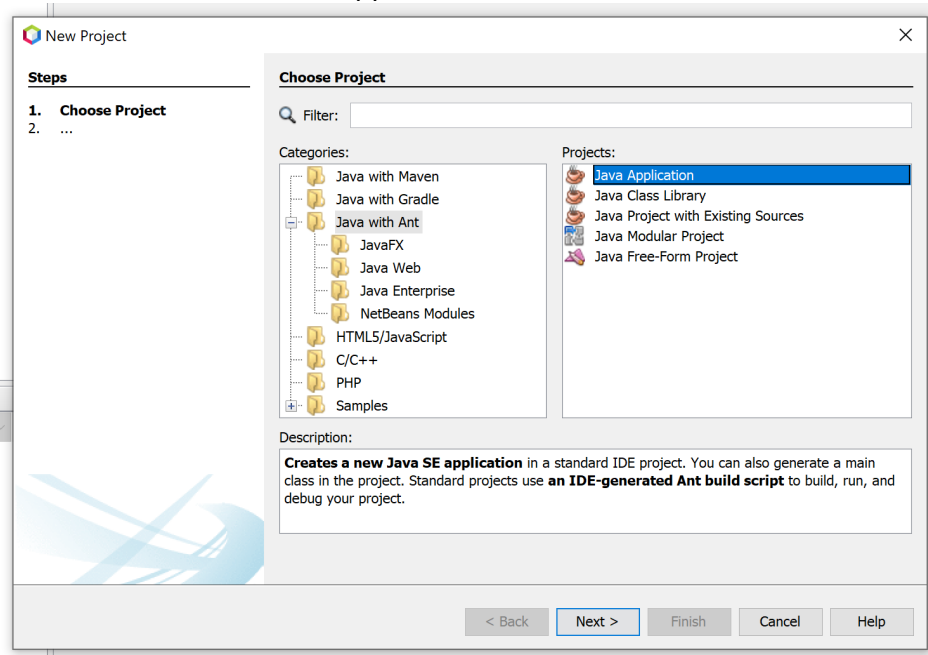
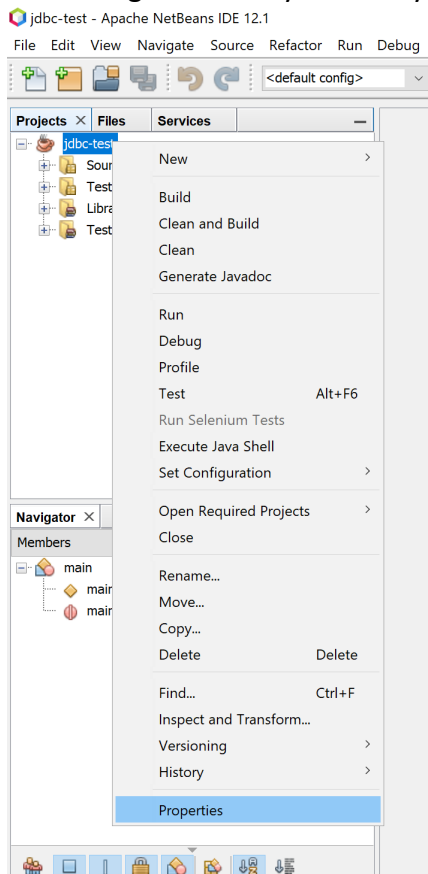


Establishing Connection

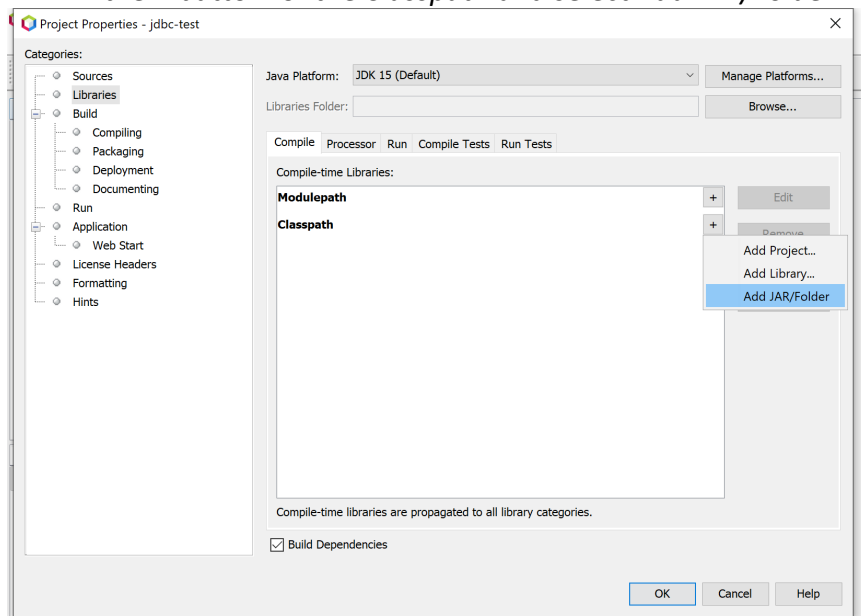
1. Create a new Java Application.



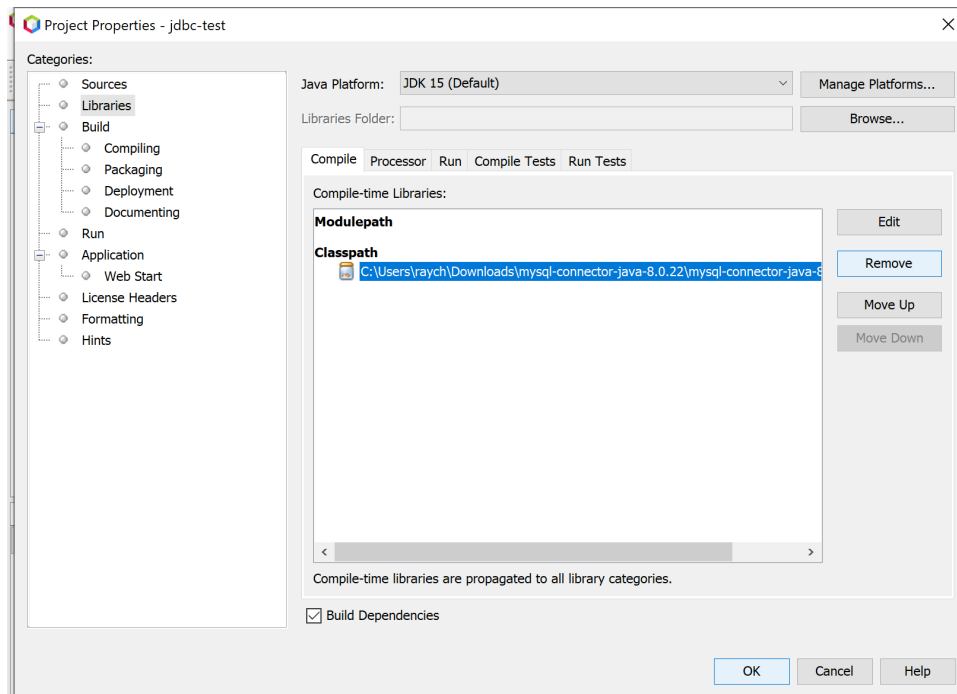
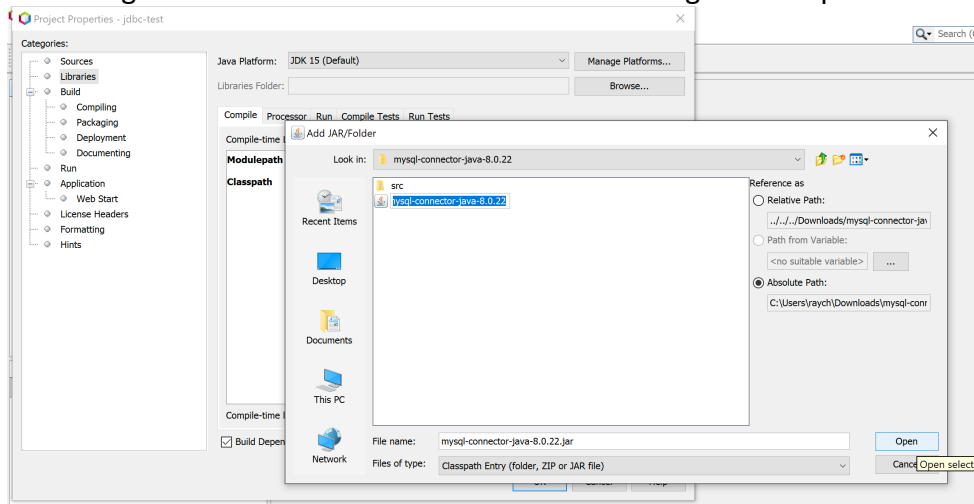
2. Right click on your newly created project and select *Properties*.



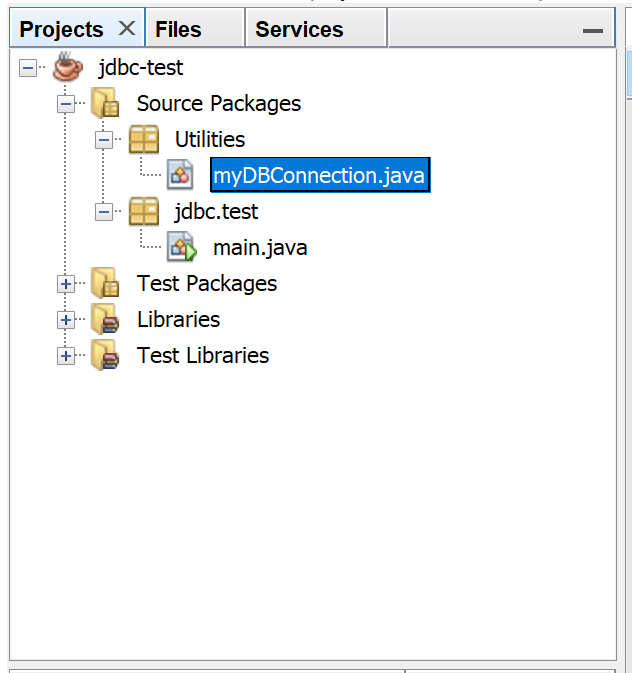
3. Go to the *Libraries* category and select the *Compile* tab. You will then need to click on the + button of the *Classpath* and select *Add JAR/Folder*.



4. Again locate the JAR file that I included during the SetUp and add it to the *Classpath*.

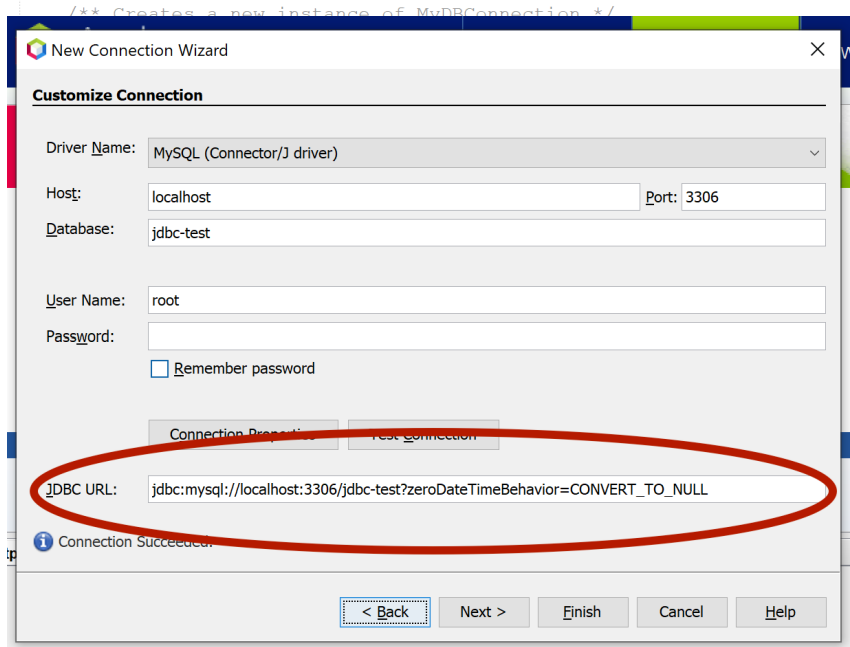


5. In the Source Packages of your project, set your main class, create a folder(*Utilities*) and add a new class(*myDBConnection*).



6. In the *myDBConnection* class, add these following variables.

```
public class myDBConnection {  
  
    private Connection myConnection;  
    private String dbUrl = "jdbc:mysql://localhost:3306/jdbc-test?zeroDateTimeBehavior=CONVERT_TO_NULL";  
    private String driver = "com.mysql.jdbc.Driver";  
    private String dbUsername = "root";  
    private String password = "";
```



You can find the url here.

7. Add this following methods in *myDBConnection*.

```
public void init() {  
  
    try{  
        Class.forName(driver);  
        myConnection=DriverManager.getConnection(dbUrl,dbUsername,password);  
        System.out.println("Successful connection");  
    }  
    catch(Exception e){  
        System.out.println("Failed to get connection");  
        e.printStackTrace();  
    }  
}
```

This method is for initializing the connection of your database to your project.

```
public Connection getMyConnection() {  
    return myConnection;  
}
```

```
public void close(ResultSet rs){  
  
    if(rs !=null){  
        try{  
            rs.close();  
        }  
        catch(Exception e){}  
    }  
}  
  
public void close(java.sql.Statement stmt){  
  
    if(stmt !=null){  
        try{  
            stmt.close();  
        }  
        catch(Exception e){}  
    }  
}
```

This method is called once you are done with an operation to prevent data leaks.

```

public void destroy() {

    if(myConnection !=null) {

        try{
            myConnection.close();
        }
        catch(Exception e){}

    }

}

```

This method is only called if you want to cut the connection of your project/application to your database.

8. Add these lines of code in your main class then try to run it. You will then see if your connection is successful or failed in the console of your NetBeans.

```

public static void main(String[] args) {
    // TODO code application logic here

    //Create instance of db class
    myDBConnection db = new myDBConnection();
    //initialize db connection
    db.init();
    db.getMyConnection();
}

```