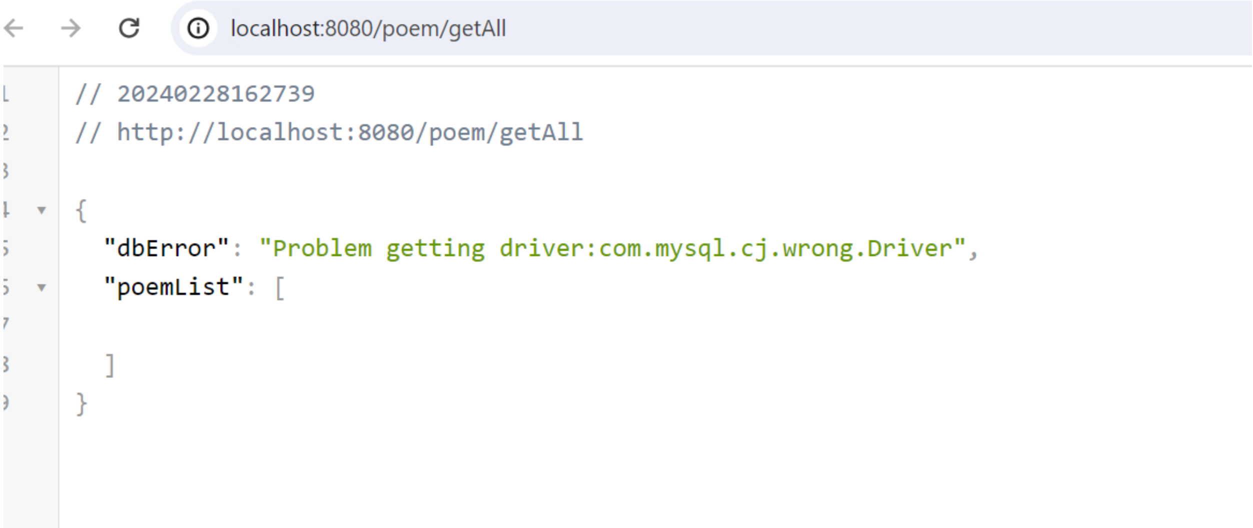
1. Missing Database Driver
   This error can be created by changing the value of DRIVER string in DbConn to a driver that doesn’t exist. It can be fixed by making sure that the DRIVER string is valid.

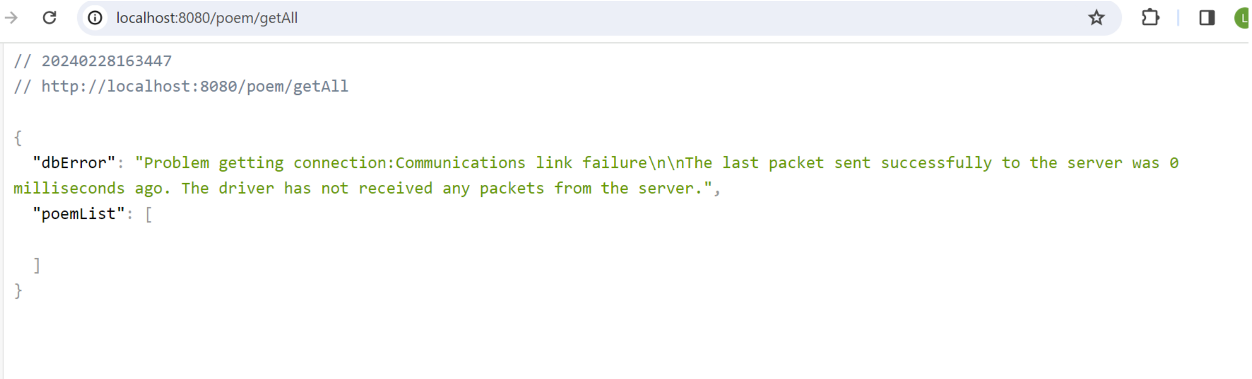


```
// 20240228162739
// http://localhost:8080/poem/getAll

{
  "dbError": "Problem getting driver:com.mysql.cj.wrong.Driver",
  "poemList": [

  ]
}
```

2. Database Unreachable
   This error can be created by changing the value of url string in DbConn to an url that doesn’t exist. It can be fixed by making sure that the url string is valid.

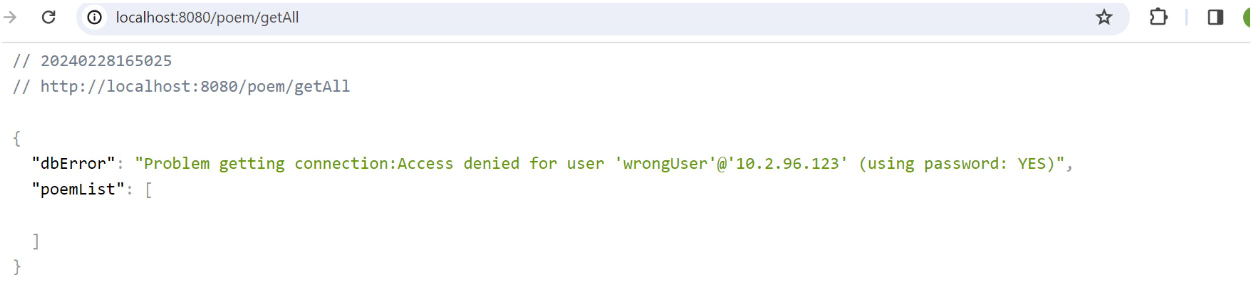


```
// 20240228163447
// http://localhost:8080/poem/getAll

{
  "dbError": "Problem getting connection:Communications link failure\n\nThe last packet sent successfully to the server was 0
milliseconds ago. The driver has not received any packets from the server.",
  "poemList": [

  ]
}
```

3. Database Not Authorized
   This error can be created by changing the value of dbAndPass string in DbConn to an user and pass that doesn’t exist. It can be fixed by making sure that the dbAndPass string is valid.



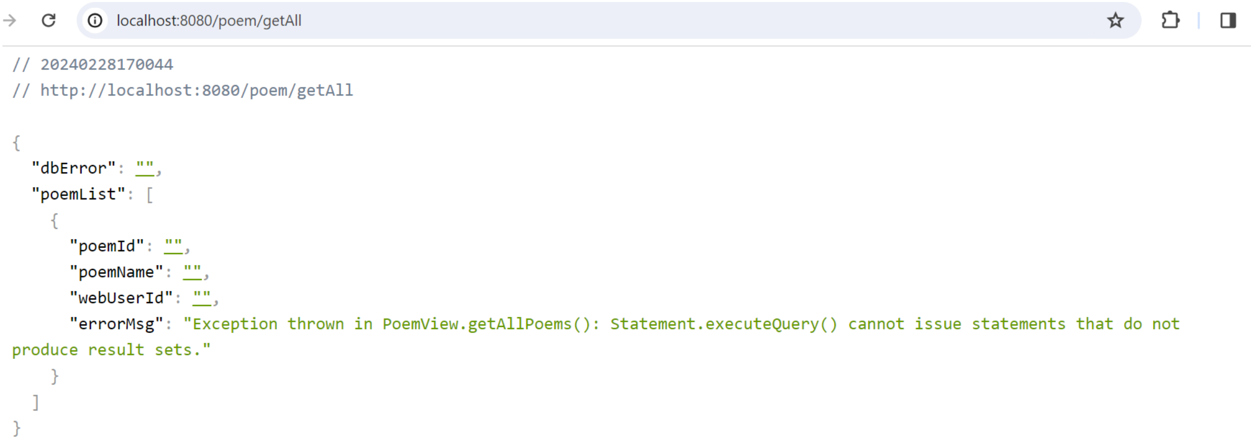
```
// 20240228165025
// http://localhost:8080/poem/getAll

{
  "dbError": "Problem getting connection:Access denied for user 'wrongUser'@'10.2.96.123' (using password: YES)",
  "poemList": [

  ]
}
```

4. Syntax error in SQL Statement

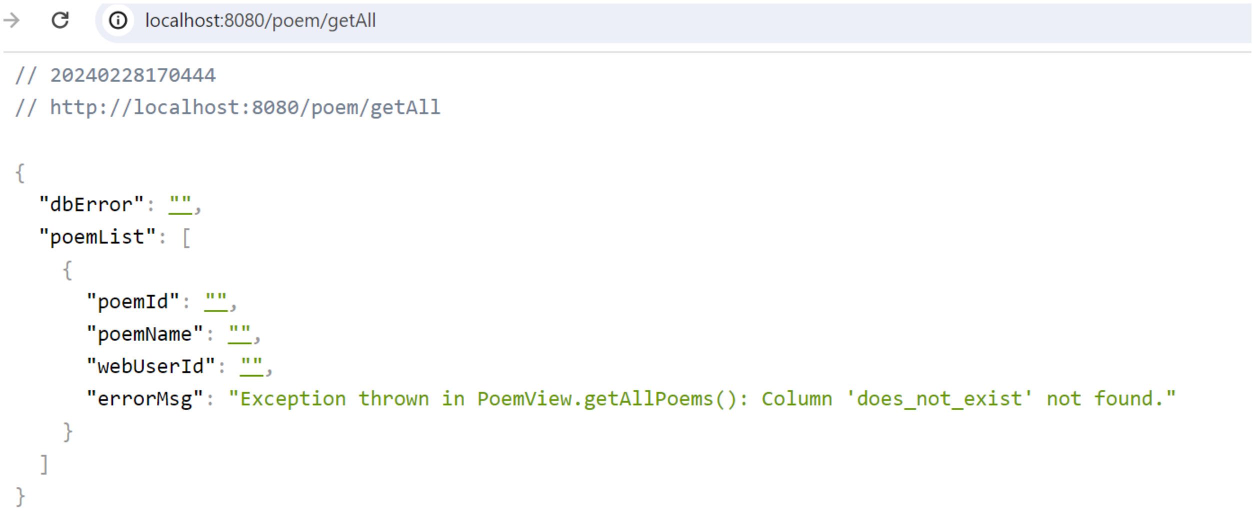
This error can be created by changing the value of sql string in PoemView (or WebUserView) to a statement that doesn't use valid syntax. It can be fixed by making sure that the sql string is valid.



```
// 20240228170044
// http://localhost:8080/poem/getAll

{
  "dbError": "",
  "poemList": [
    {
      "poemId": "",
      "poemName": "",
      "webUserId": "",
      "errorMsg": "Exception thrown in PoemView.getAllPoems(): Statement.executeQuery() cannot issue statements that do not
produce result sets."
    }
  ]
}
```

5. Error Extracting Data from Result Set (bad column name)
   This error can be created by changing the value of columnLabel argument string in PoemView (or WebUserView), within "sd.xxx" to a column that doesn't exist. It can be fixed by making sure that the columnLabel string is valid.

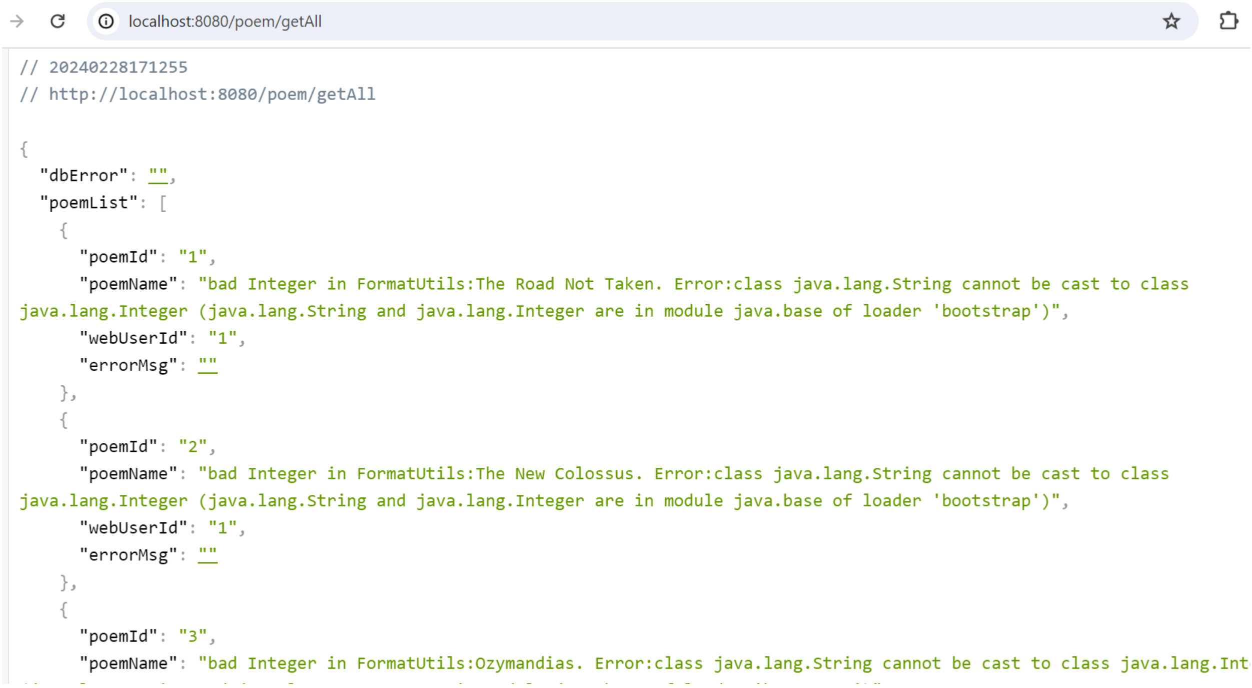


localhost:8080/poem/getAll

```
// 20240228170444
// http://localhost:8080/poem/getAll

{
  "dbError": "",
  "poemList": [
    {
      "poemId": "",
      "poemName": "",
      "webUserId": "",
      "errorMsg": "Exception thrown in PoemView.getAllPoems(): Column 'does_not_exist' not found."
    }
  ]
}
```

6. Error Extracting Data from Result Set (wrong data type)
   This error can be created attempting to format a string as integer in PoemView or WebUserView, it can be fixed by formatting strings as strings with the proper method from FormatUtils.



localhost:8080/poem/getAll

```
// 20240228171255
// http://localhost:8080/poem/getAll

{
  "dbError": "",
  "poemList": [
    {
      "poemId": "1",
      "poemName": "bad Integer in FormatUtils:The Road Not Taken. Error:class java.lang.String cannot be cast to class
java.lang.Integer (java.lang.String and java.lang.Integer are in module java.base of loader 'bootstrap')",
      "webUserId": "1",
      "errorMsg": ""
    },
    {
      "poemId": "2",
      "poemName": "bad Integer in FormatUtils:The New Colossus. Error:class java.lang.String cannot be cast to class
java.lang.Integer (java.lang.String and java.lang.Integer are in module java.base of loader 'bootstrap')",
      "webUserId": "1",
      "errorMsg": ""
    },
    {
      "poemId": "3",
      "poemName": "bad Integer in FormatUtils:Ozymandias. Error:class java.lang.String cannot be cast to class java.lang.Int
```