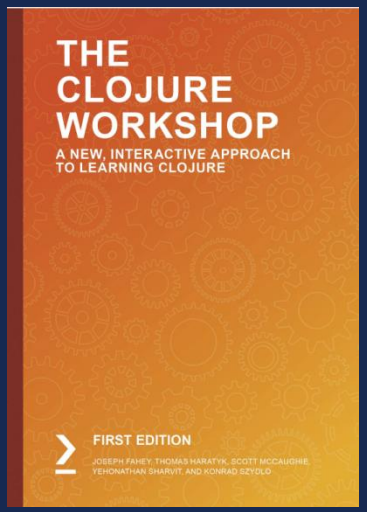
Programação Funcional

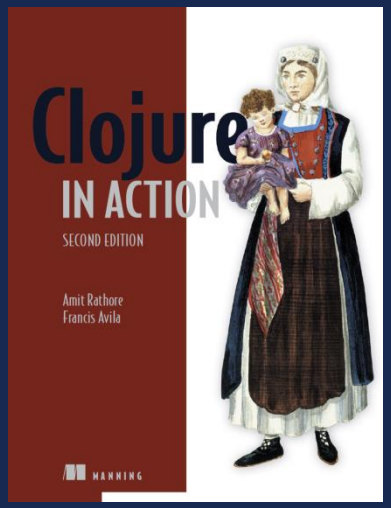
Unidade 11 – Interação com Banco de Dados

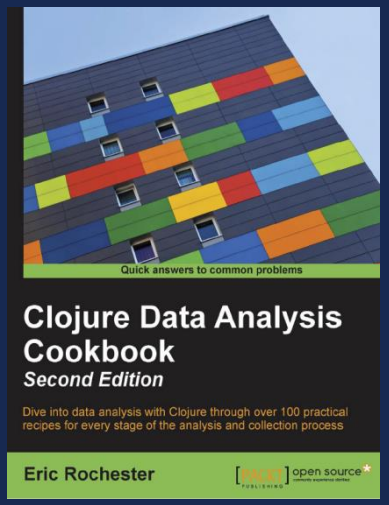
Prof. Aparecido V. de Freitas
Doutor em Engenharia
da Computação pela EPUSP
aparecido.freitas@prof.uscs.edu.br
aparecidovfreitas@gmail.com

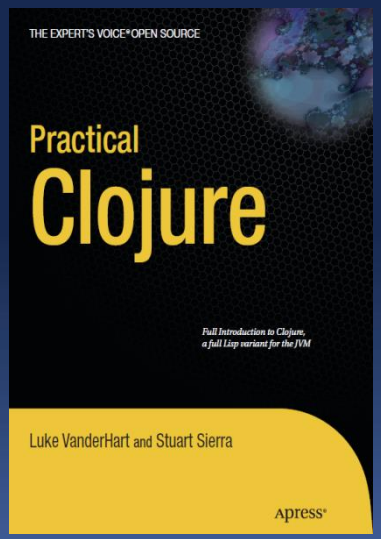
Revisão Técnica: Maurício Szabo
mauricio.szabo@gmail.com

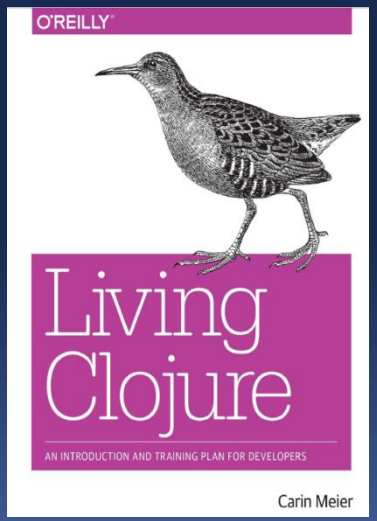
Bibliografia

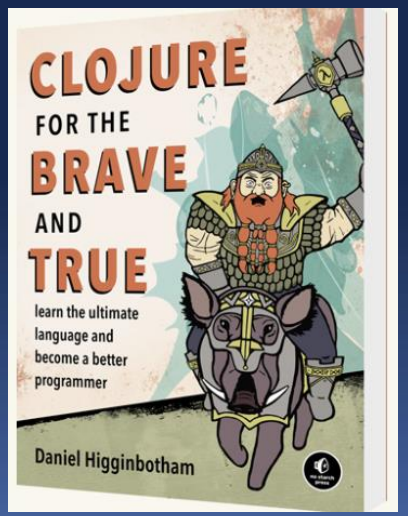












Introdução

- Cada vez que reiniciamos nosso **REPL**, perdemos os dados manipulados previamente;
- Veremos nessa unidade a interação de Clojure com sistemas de bancos de dados relacionais;
- Utilizaremos, por simplicidade, o **RDBMS Apache Derby**, implementado inteiramente em **Java**, proporcionando dessa forma mais tempo para focar em aspectos de modelagem de dados, persistência e recuperação.

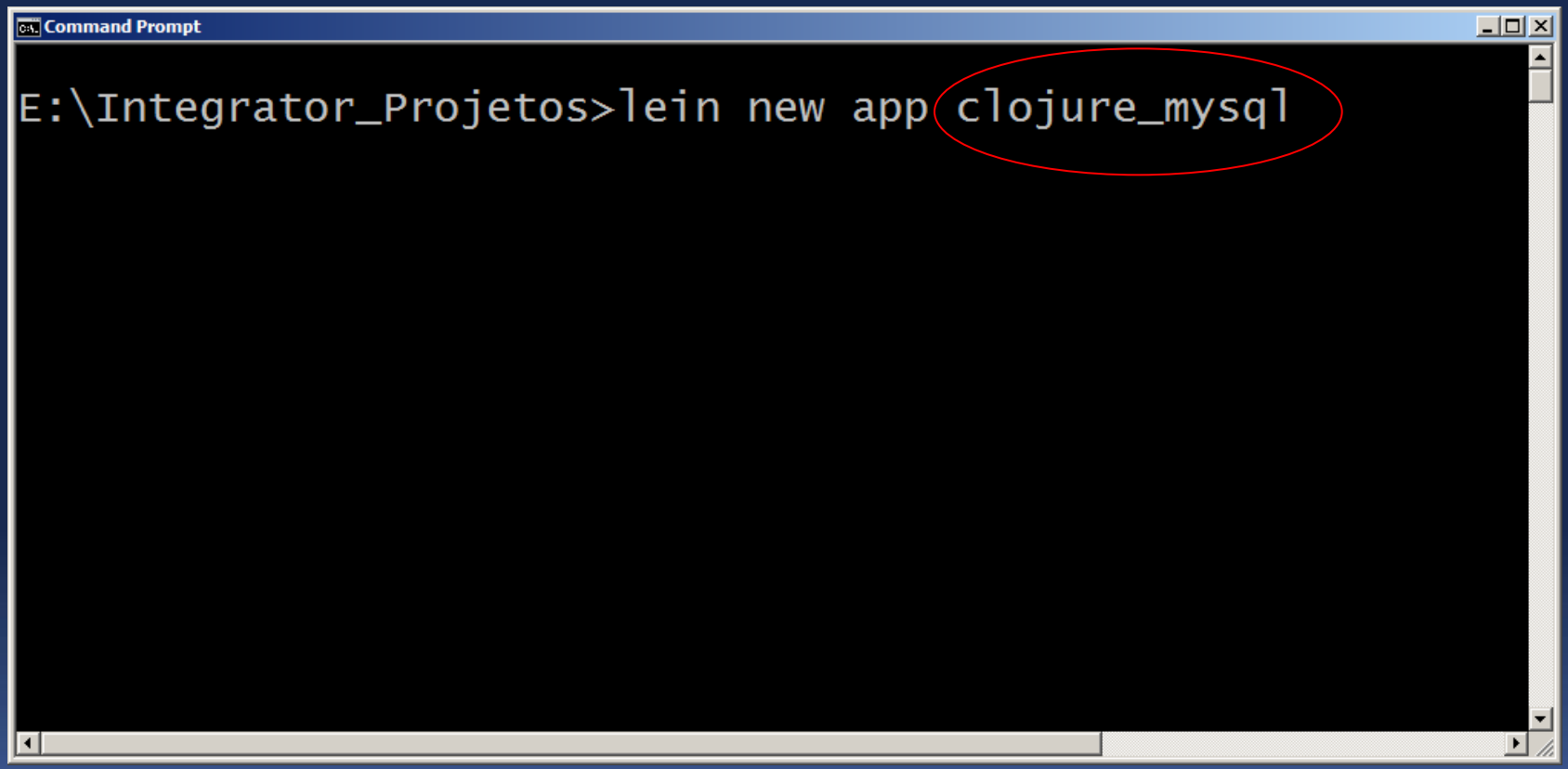
Conectando ao Banco de Dados

- Em termos de interação com o banco de dados Apache **Derby** a partir da linguagem Clojure, utilizaremos a **Clojure.java.jdbc**, um biblioteca de baixo nível para interagir com o Banco de Dados, por meio da **JDBC**.

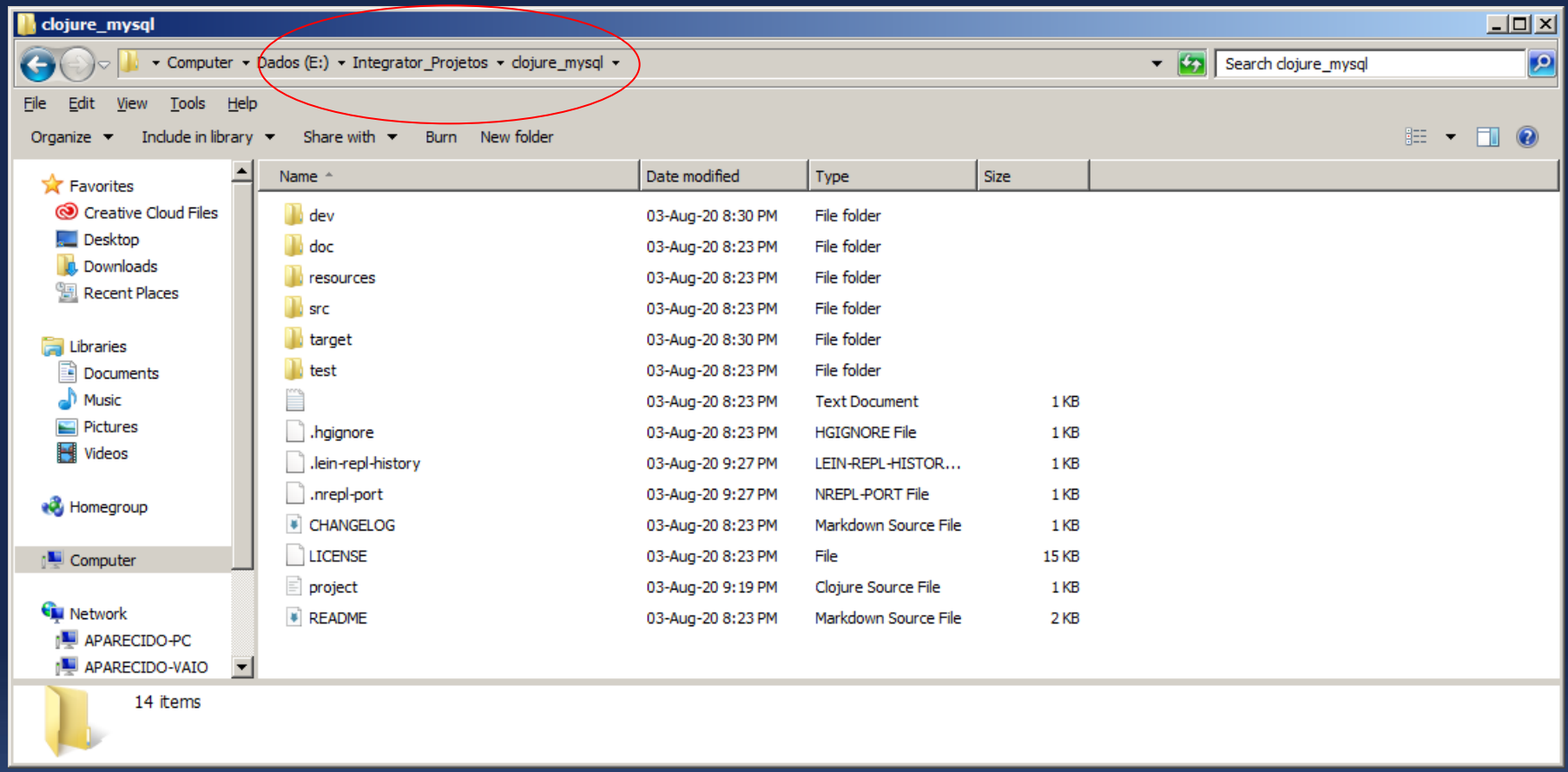
Acessando MySQL com projeto Leiningen

Recursos

- Criando o projeto com Lein



Projeto criado



Configurando o projeto

- Incluindo no arquivo **project.clj** as dependências do **jdbc** e do Driver jdbc **MySQL**

project.clj

```
(defproject clojure_mysql "0.1.0-SNAPSHOT"
  :description "FIXME: write description"
  :url "http://example.com/FIXME"
  :license {:name "EPL-2.0 OR GPL-2.0-or-later WITH Classpath-exception-2.0"
            :url "https://www.eclipse.org/legal/epl-2.0/"}

  :repl-options {:init-ns user}

  :dependencies [ [org.clojure/clojure "1.10.1"]
                  [org.clojure/java.jdbc "0.7.9"]
                  [mysql/mysql-connector-java "8.0.21"]]

  :jvm-opts ["-Dclojure.server.myrepl={:port,5555,:accept,clojure.core.server/repl}"]

  :main ^:skip-aot clojure_mysql.core
  :target-path "target/%s"
  :profiles {:uberjar {:aot :all
                       :jvm-opts ["-Dclojure.compiler.direct-linking=true"] }
                       :dev {:dependencies []
                       :source-paths ["dev"]}})
```

Criando o Banco de Dados

- Banco de Dados: scpe
- Porta: 3307
- Host: localhost
- Tabela: cursos

```
use scpe;

describe curso;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
id_curso	int(11)	NO	PRI	NULL	auto_increment
datetime_curso	varchar(255)	YES		NULL	
nome_curso	varchar(255)	YES		NULL	
timestamp_curso	varchar(255)	YES		NULL	

```
select *  from curso;
```

Result Grid | Filter Rows: | Edit: | Export/Import:

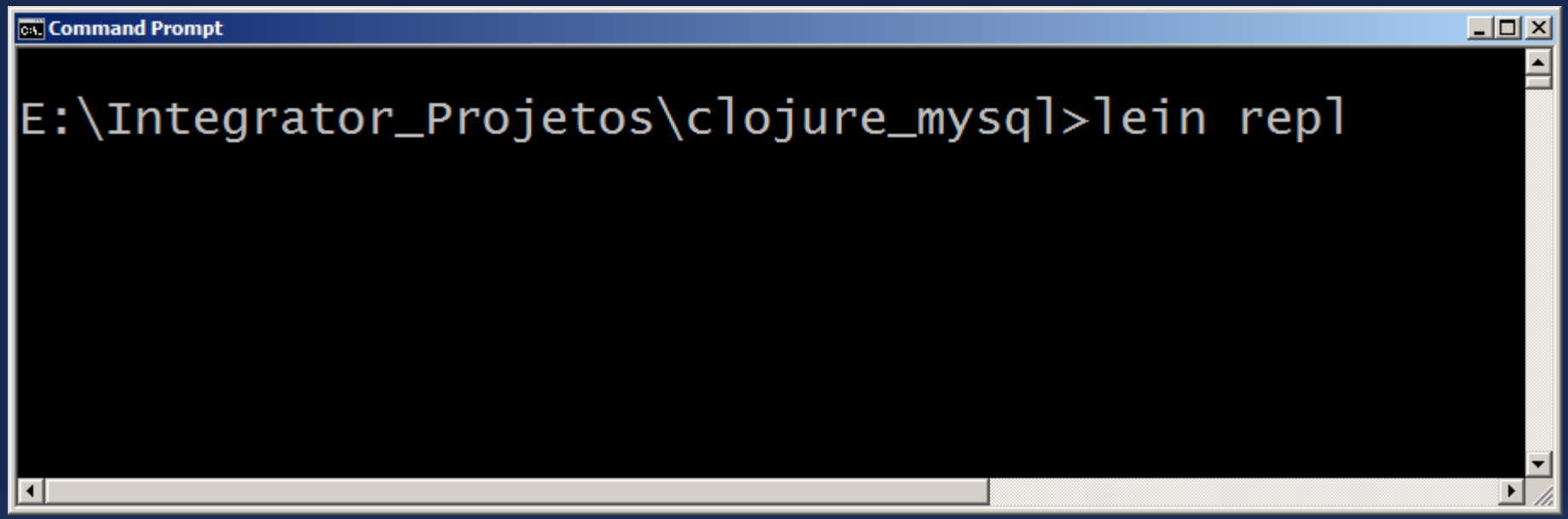
id_curso	datetime_curso	nome_curso	timestamp_curso
1	2020-07-20 21:51:15	Sistemas de Informação	2020-07-20 21:51:15
2	2020-07-20 21:51:18	Gestão de Tecnologia da Informação	2020-07-20 21:51:18
3	2020-07-20 21:51:21	Tecnologia ADS	2020-07-20 21:51:21

Código Clojure para acessar a tabela

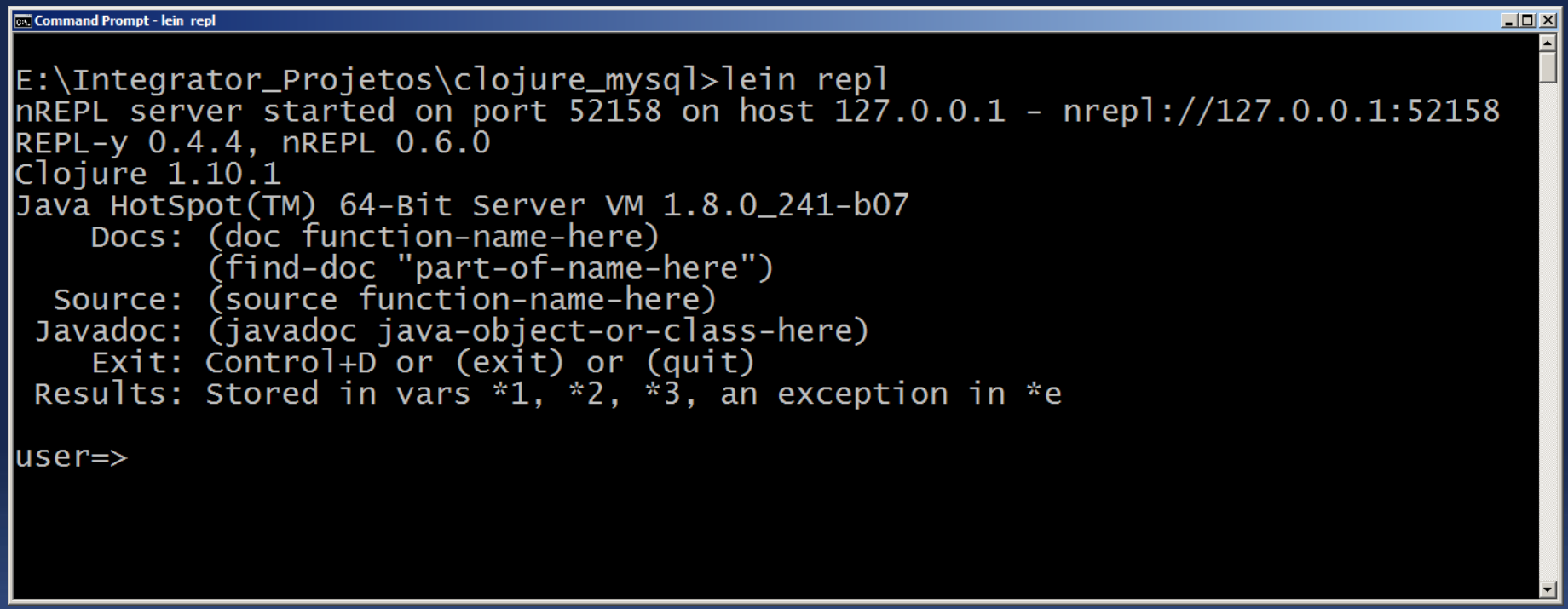
core.clj

```
(ns clojure_mysql.core
  (:gen-class))

(require '[clojure.java.jdbc :as jdbc])

(let [db-host "localhost"
      db-port 3307
      db-name "scpe"]

(def db {:classname "com.mysql.cj.jdbc.Driver"
         :subprotocol "mysql"
         :subname (str "//" db-host ":" db-port "/" db-name "?useTimezone=true&serverTimezone=UTC")
         :user "root"
         :password "maua"}))

(def resp (jdbc/query  db ["SELECT * FROM curso WHERE id_curso = 1"]))

(defn -main
  "Código clojure para acessar MySQL via jdbc"
  [& args]
  (println resp))
```

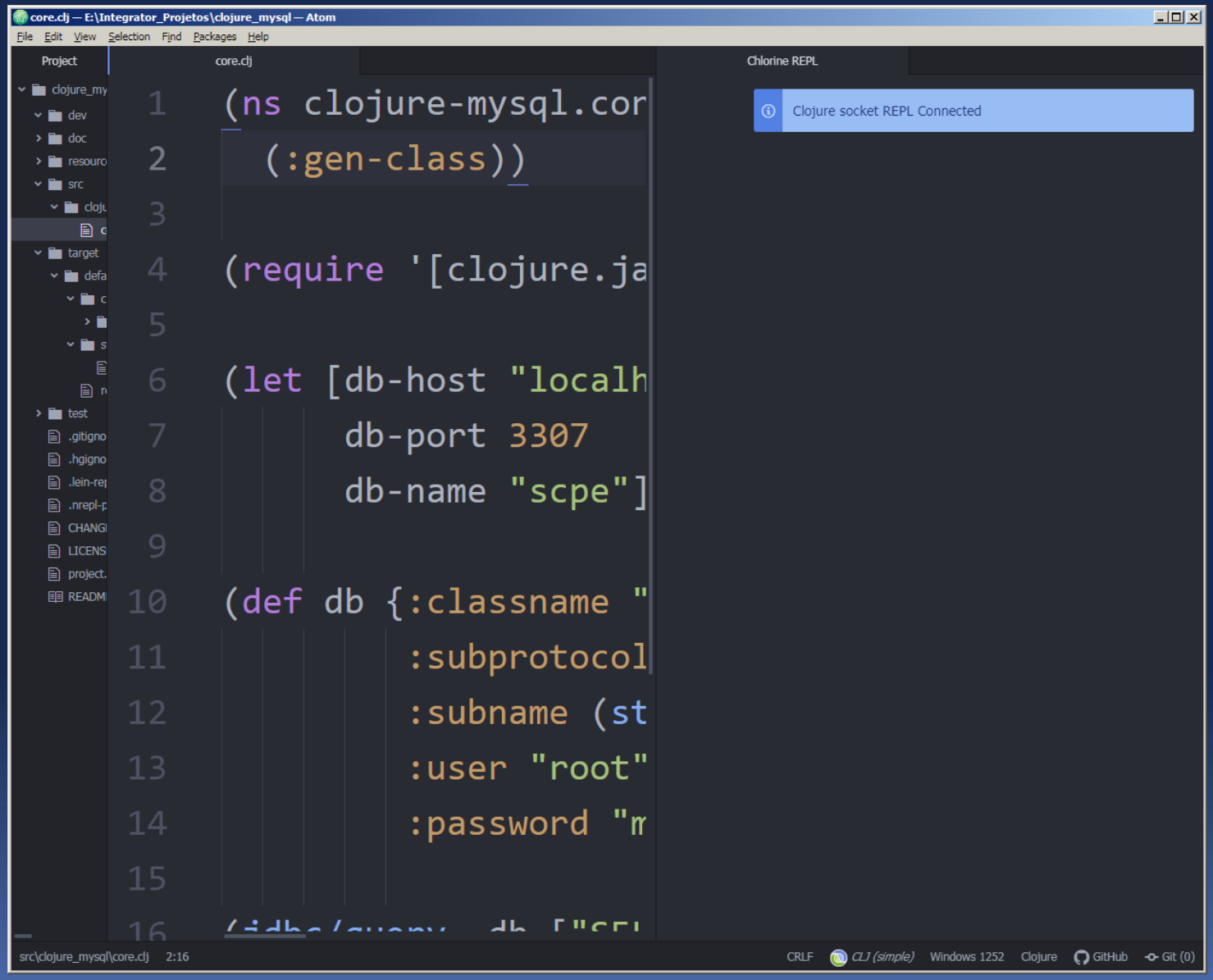
Executando sob REPL - Chlorine



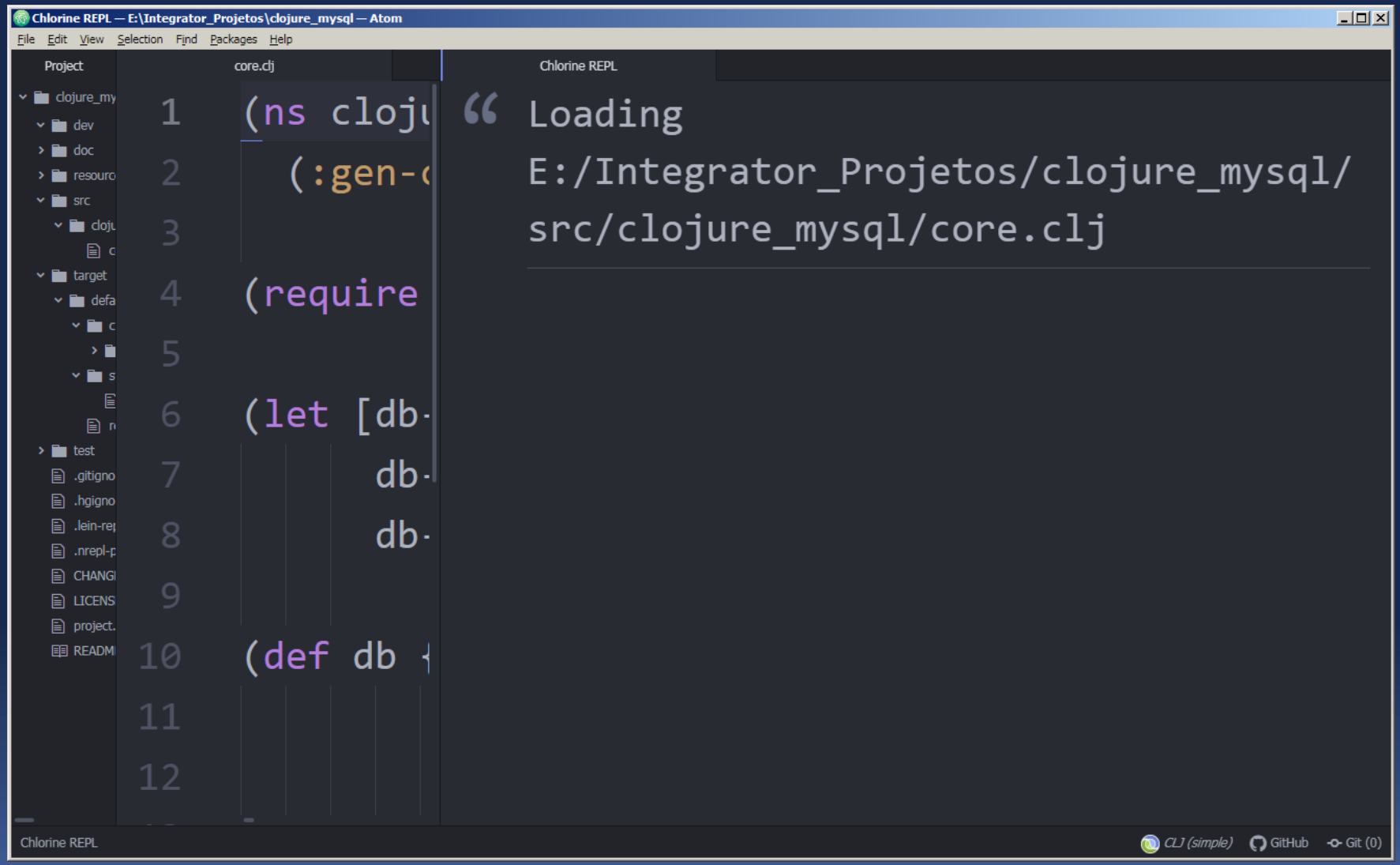
Servidor REPL ativo



Conectando Atom Chlorine



Carregando código no REPL (Load-file)



Executando sob REPl (Chlorine)

core.clj — E:\Integrator_Projetos\clojure_mysql — Atom

File Edit View Selection Find Packages Help

Project

core.clj

```
(def db {:classname "com
         :subprotocol "m
         :subname (str "
         :user "root"
         :password "maua

(jdbc/query  db ["SELECT
  ({:id_curso 1, :datet

(defn -main
  "Código clojure para a
  [& args]
```

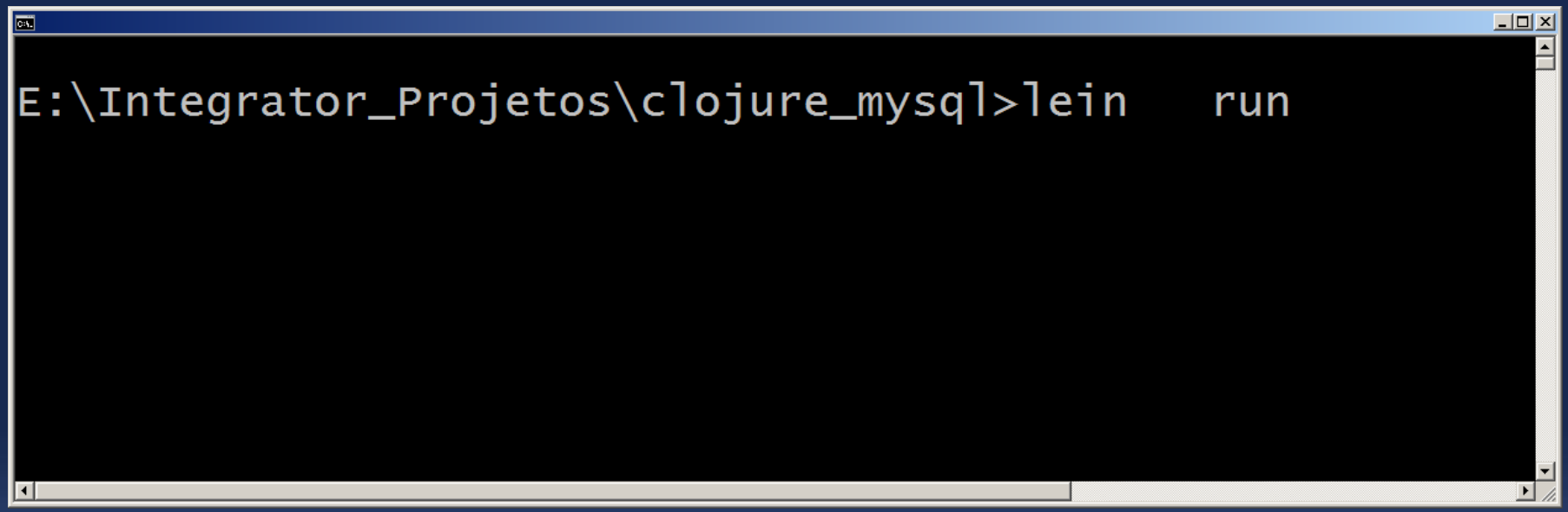
Chlorine REPL

```
Loading
E:/Integrator_Projetos/clojure_mysq
l/src/clojure_mysql/core.clj
({:id_curso 1, :datetime_curso "2020
```

src\clojure_mysql\core.clj 4:6 CRLF CLJ (simple) Windows 1252 Clojure GitHub Git (0)

Executando – Linha de Comando REPL

No diretório do projeto > lein run



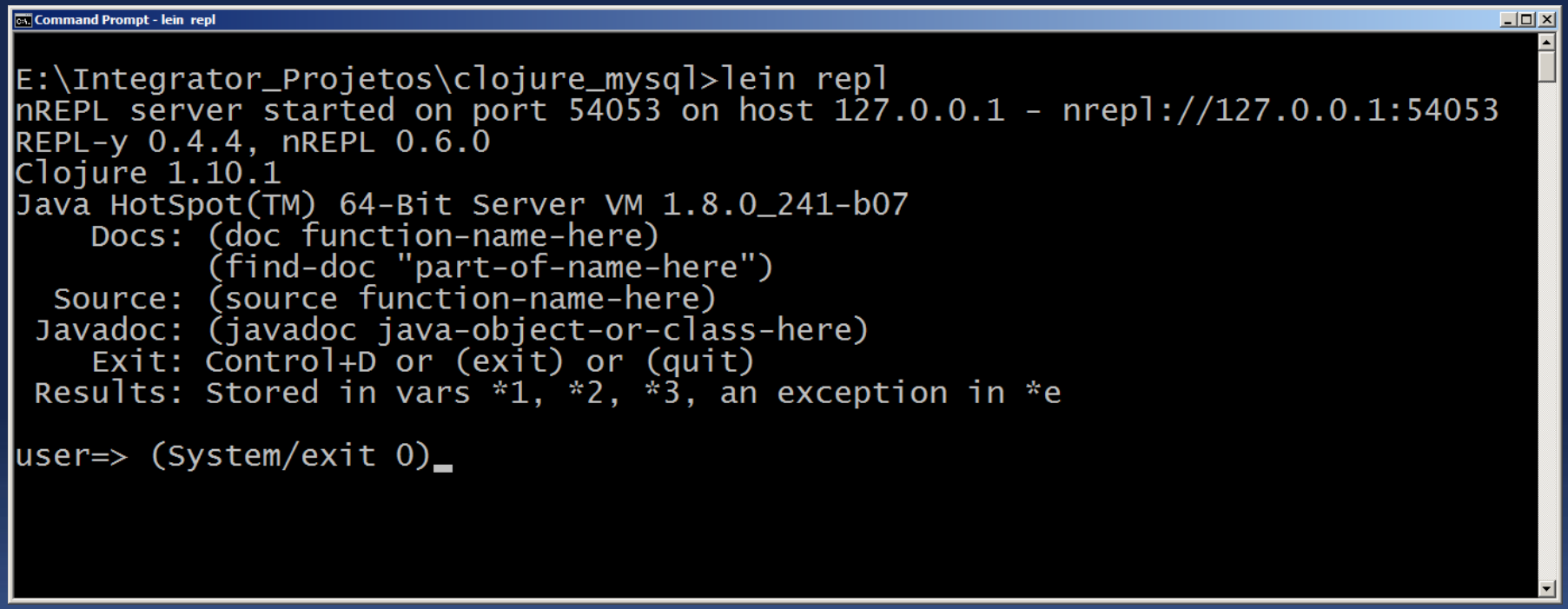
Executando – Linha de Comando REPL

- Caso, ocorra o erro "Address already in use: JVM_Bind, será necessário efetuar shutdown no repl server ativo.

```
E:\Integrator_Projetos\clojure_mysql>lein   run
Exception in thread "main" java.net.BindException: Address already in use: JVM_B
ind
        at java.net.DualStackPlainSocketImpl.bind0(Native Method)
        at java.net.DualStackPlainSocketImpl.socketBind(DualStackPlainSocketImpl
.java:106)
        at java.net.AbstractPlainSocketImpl.bind(AbstractPlainSocketImpl.java:38
7)
        at java.net.PlainSocketImpl.bind(PlainSocketImpl.java:190)
        at java.net.ServerSocket.bind(ServerSocket.java:375)
        at java.net.ServerSocket.<init>(ServerSocket.java:237)
        at clojure.core.server$start_server.invokeStatic(server.clj:101)
        at clojure.core.server$start_servers.invokeStatic(server.clj:160)
        at clojure.core.server$start_servers.invoke(server.clj:157)
        at clojure.lang.Var.invoke(Var.java:384)
        at clojure.lang.RT.doInit(RT.java:493)
        at clojure.lang.RT.init(RT.java:467)
        at clojure.main.main(main.java:38)
```

Executando – Linha de Comando REPL

- Para efetuar shutdown no REPL server => (System/exit 0)



Executando – Linha de Comando REPL