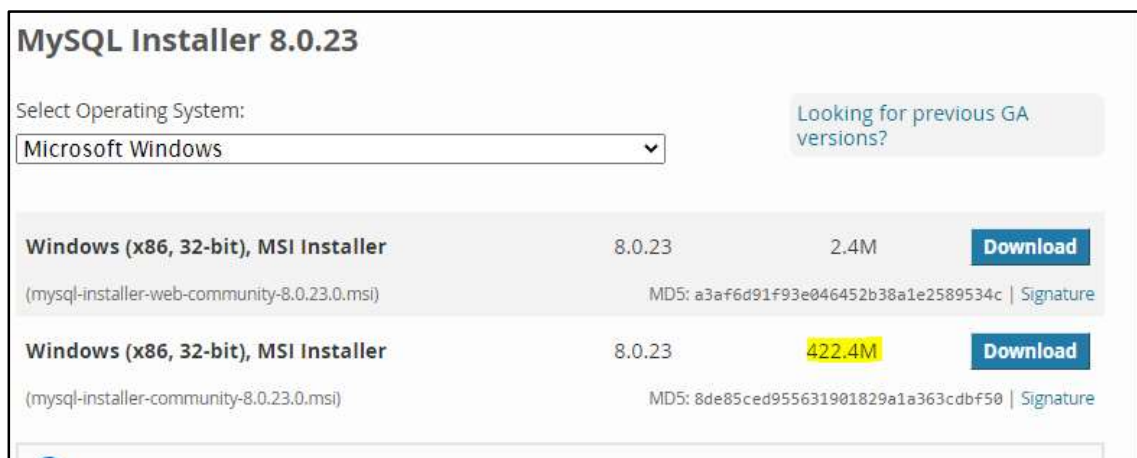


# DOCUMENTAÇÃO 1

Thiago Fernandes Canonici

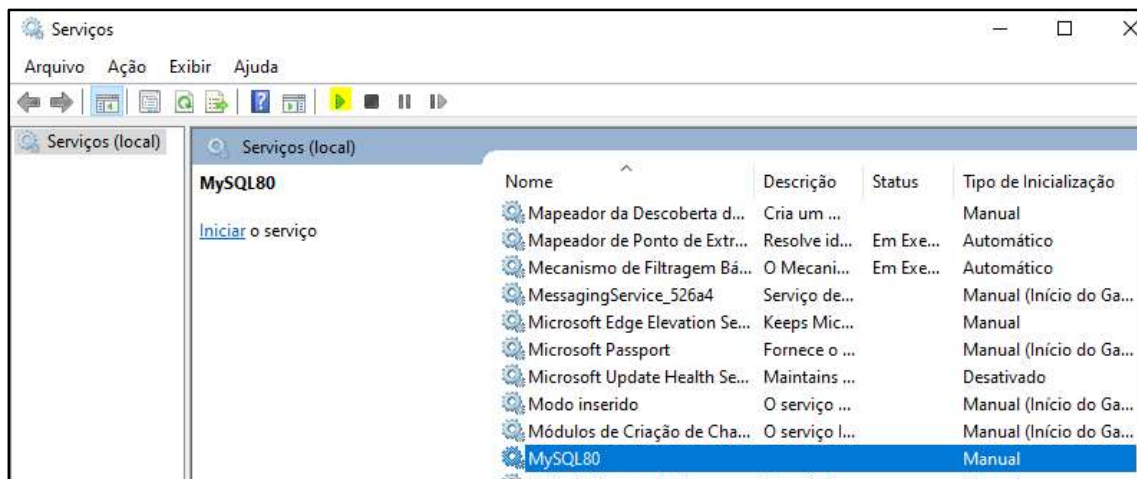
## MANUAL

O banco de dados utilizado foi o MySQL. Para instalá-lo vá em <https://dev.mysql.com/downloads/> escolha “MySQL Installer for Windows” e faça o download do programa mais pesado, porque ele conterá todos os arquivos necessários sem haver necessidade da instalação posterior deles (Fig. 1).



**Figura 1:** Após clicar em “MySQL Installer for Windows” escolher a segunda opção.

Após a instalação do banco de dados MySQL, Na barra de inicialização, digite “ferramentas administrativas”, duplo clique em “Serviços” e clique em iniciar caso o status não apareça “Em Execução” (Fig. 2).

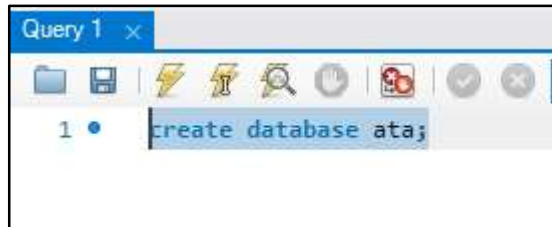


**Figura 2:** verificar em “Serviços” se o MySQL está em execução, caso contrário clicar no ícone de inicializar serviço destacado em amarelo.

Vá em MySQL Workbench instalado no seu computador e, na área de trabalho do banco de dados, digite “create database nomeDaTabela;”, selecione a linha e clique no raio para criar. Para aparecer o banco de dados criado, atualize a guia “SCHEMAS”. Duplo clique no banco de dados criado (o nome deve aparecer em negrito) e vá na barra de ferramentas do MySQL Workbench, clicar em *File* e *Open SQL Script...* e carregar o arquivo da figura 4 salvo com o nome “database.sql”. O Script SQL criará uma tabela funcionario<sup>1</sup> com as colunas Id, Participante, Area<sup>2</sup>, Email e Telefone. Selecione todo o Script e clique no raio.

<sup>1</sup> O nome foi deixado sem acentuação para não ter problema de reconhecimento da palavra pelo MySQL;

<sup>2</sup> Idem ao item 1.



**Figura 3:** criação do banco de dados “ata”. Digite “create database ata;”, selecione a linha e clique no raio na barra de ferramentas da guia “Query 1”. O banco de dados será criado.

```
CREATE TABLE funcionario (
  Id int(11) NOT NULL AUTO_INCREMENT,
  Participante varchar(300) NOT NULL,
  Area varchar(300) NOT NULL,
  Email varchar(300) NOT NULL,
  Telefone varchar(200) NOT NULL,
  PRIMARY KEY (Id)
);
```

**Figura 4:** Script SQL criando a tabela funcionario.

Pesquisar “mysql java conector” para instalá-lo, escolha “Plataform Independent” e faça o download do “Zip Archive”, na página seguinte não é obrigado a fazer o Login; clique em “No thanks (...)”. Será utilizada a biblioteca com a extensão *jar* (.jar), e não a que tiver *bin*. Vá no Eclipse ou Spring em Window/Preference/Java/Build Path/User Libraries vá em New de um nome a sua escolha, no caso “MySQL Connector”, selecione a pasta criada e clique em *Add External JARs...* e localize o arquivo com extensão *jar* que foi baixado, abra-o e clique em *Apply and Close*.

Crie um *Java Project* no Eclipse escolha um nome para o projeto e clique em “Next>”. Acesse a guia *Libraries*, selecione o *Modulepath* clique em *Add Library...* e *User Library*, selecione a pasta da biblioteca criada, no caso “MySQL Connector” e *Finish* clicar em *Finish* novamente e não crie um Módulo.

Clique com o botão direito na pasta do projeto vá em *New/File*. Dar o nome de “db.properties” e clicar em *Finish*. Copie o código da figura 5, atentando-se para os nomes de usuário, senha e nome do banco de dados corretos<sup>3</sup>. O nome criado para o banco de dados foi feito anteriormente com o comando “create database nomeDoArquivo;” (Fig. 3).

```
1 user=Thiago
2 password=1234
3 dburl=jdbc:mysql://localhost:3306/ata
4 useSSL=false
```

**Figura 5:** Arquivo com os dados do banco de dados. Atente-se para o campo *user=*, *password=*, e o final do *dburl=* que aqui está “ata”. Os nomes devem ser o seu usuário, a sua senha e o nome dado ao seu banco de dados com o comando “create database nomeDoArquivo;” em passos passado.

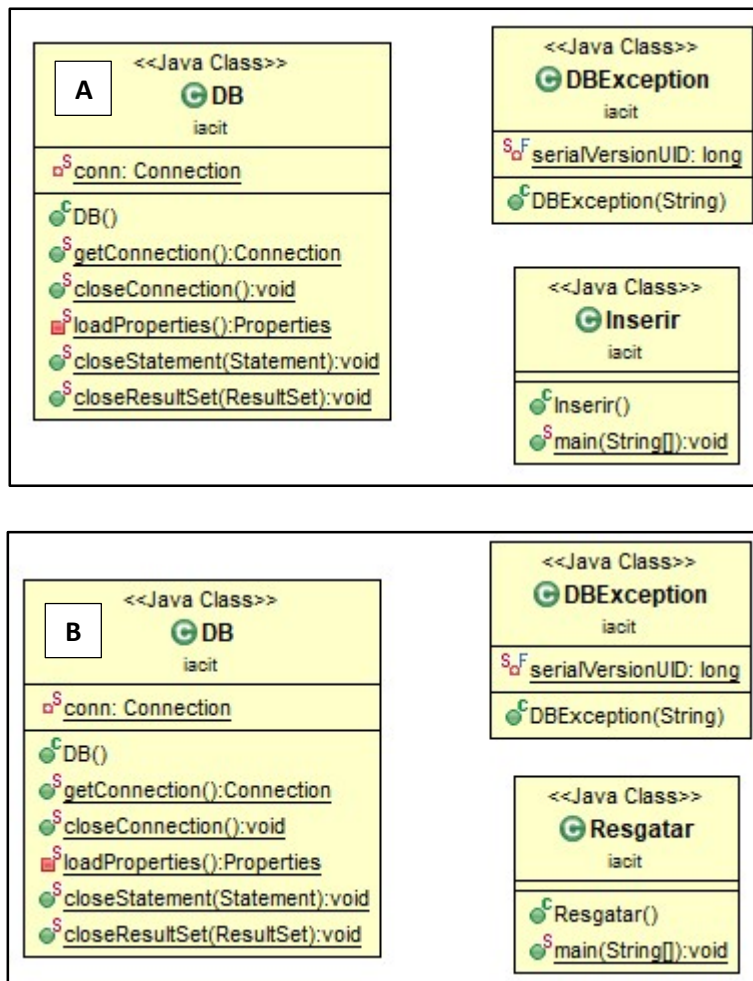
---

<sup>3</sup> Lembre-se dos seus dados, a figura 5 não funcionará caso os campos usuário, senha e dburl estiverem com dados que não são seus.

## FUNCIONALIDADE

A funcionalidade para a *sprint 1* foi a inserção e exibição dos participantes da ata. Por enquanto a exibição está sendo armazenada em um arquivo *csv*, mas que posteriormente será exibido em *xls* e *pdf*.

A UML simplifica o código criado. Na classe DB, tem-se as implementações dos métodos `getConnection`, `closeConnection`, `loadProperties`, `closeStatement` e `closeResultSet`. Na inserção dos dados foram utilizados o `PreparedStatement` e o `executeUpdate` (Fig. 6A). O objeto *PreparedStatement* permite montar o comando SQL e deixa os parâmetros serem postos depois. O `executeUpdate` consolida a inserção dos dados declarados; no comando de recuperação dos dados foram utilizadas as classes *Statement* e *ResultSet*. A classe *Statement* aceita um comando SQL para ser executado. A classe *ResultSet* é um objeto que armazenará o resultado da consulta SQL, armazenando os dados em forma de tabela (Fig.6B).



**Figura 6:** UML das funcionalidades de inserção e resgate das informações dos participantes da ata. (A) UML da inserção dos dados; (B) UML do resgate dos dados.