

PECULIARIDADES DE UM BANCO MYSQL

Gustavo Minuzzi Kloh¹

Taiane Tais Habitzreiter²

Velcir Barcaroli³

Resumo

Quando se trata de um manuseio e gestão de banco de dados podemos estar elencando um banco de dados denominado MySql, onde abrange sua arquitetura de código aberto, possibilitando uma linguagem estruturada e complexa do contexto. A boa escolha de um banco de dados acarreta diretamente no andamento e sucesso de um projeto, devendo elencar as prioridades necessárias para tal, deve-se lembrar que o banco MySql, é, de certa forma, conhecido como uma estrutura de fácil utilização, contribuindo com sua interface simples e com compatibilidade vasta com diversos sistemas operacionais.

Palavras-chave: MySql; Projeto; Sistema; Banco.

Abstract

When it comes to database handling and management, we may be listing a database called MySql, which covers its open source architecture, enabling a structured and complex language in the context. The good choice of a database directly affects the progress and success of a project, and must list the priorities necessary for this, it should be remembered that the MySql database is, in a way, known as an easy-to-use structure, contributing with its simple interface and wide compatibility with different operating systems.

Keywords: MySql; Project; System; Bank.

INTRODUÇÃO

O banco de dados MySql é conhecido como uma linguagem de consulta de informações, onde tem seu padrão de uso para manipulação de bases de dados que trabalham de forma relativa, esse banco atua diretamente com diversas empresas que atuam com um volume maior de dados ou informações, podemos citar algumas de exemplo onde o banco de dados MySql está inserido que é a NASA, Bradesco, Sony e HP. Podemos notar que todas as empresas citadas atuam com um número gigantesco de informação diariamente e necessitam de um bom e eficaz gerenciador de banco de dados.

¹ Graduado em Gestão da Tecnologia da Informação da FAI Faculdades de Itapiranga e pós-graduado em Tecnologia da Informação na universidade Faveni. Professor no Centro Universitário FAI. Analista de TI – Uceff. E-mail: guhkloh40@gmail.com.

² Graduada em Gestão da Tecnologia da Informação. Especialista em Educação a Distância Gestão e Tutoria. Especialista em Docência para Educação Profissional. Mestranda em Educação. Universidade Federal da Fronteira Sul. Chapecó-SC. E-mail: taianetais@hotmail.com

³ Graduado em Ciências da Computação, Especialista em Computação e Mestre em Computação Aplicada. E-mail: velcir@uceff.edu.br

O banco de dados MySQL atua com alguns recursos importantes, como a linguagem de definição de dados, a linguagem de manipulação de dados, a linguagem de consulta de dados, a linguagem de controle de dados e a linguagem de transação de dados, todos respectivamente, essenciais para o sucesso do banco citado.

Fundamentação Teórica

Atualmente o MySQL é um dos maiores sistemas de gerenciamento de banco de dados do mundo, através de pesquisas notou-se que já houve em torno de mais de 10 milhões de instalações entre usuários.

Podemos elencar diversos fatores que contribuem diretamente com o sucesso e crescimento desse banco de dados, podemos citar alguns benefícios desse SGBD:

- Portabilidade – atende e é compatível com qualquer plataforma da atualidade;
- Compatibilidade – compatível com linguagens de programação, como Java, Delphi, C, C++, C#, Python, entre outras inúmeras;
- Interface Gráfica – possui interface MySQL Toolkit;
- Excelente desempenho funcional;
- Estabilidade em sua arquitetura;
- Fácil replicação configurável;
- Compatível com Triggers;
- Software livre baseado na GPL;
- Atende arquiteturas de Functions e Stored procedures.

Criação de um banco de dados

Para início de criação de um novo banco de dados devemos utilizar a seguinte sintaxe para desenvolvimento: `CREATE DATABASE bd_teste;` Após realizar a criação do banco conforme citado devemos sinalizar ao banco que iremos fazer uso do mesmo através da seguinte sintaxe: `USE bd_teste;`

Utilizando essas duas sintaxes deixamos nosso banco de dados criado e pronto para receber novas tabelas para uso.

Criação de tabelas

Para desenvolvimento de novas tabelas ao nosso banco de dados devemos realizar a criação da seguinte sintaxe:

```
CREATE TABLE fornecedor (  
    codigo                int(4)  
    AUTO_INCREMENT,     nome  
    varchar(30) NOT NULL, email  
    varchar(50),  
    PRIMARY KEY (codigo)  
);
```

Conforme desenvolvido os comandos de “AUTO_INCREMENT” serve para que o código do registro receba de forma automática um valor quando se cria um novo registro, o comando “PRIMARY KEY” serve para definir qual será a chave primária da tabela, esse campo deve servir como chave da tabela e não pode ser repetido e o comando “NOT NULL” serve basicamente para definir que um atributo seja preenchido de forma obrigatória.

Método INSERT

Para inserção de novos registros em nossa tabela e nosso banco devemos utilizar as seguintes sintaxes para tal:

```
INSERT INTO fornecedor (codigo, nome, email) VALUES (null, “Leonardo”,  
“leo_41@gmail.com”);
```

```
INSERT INTO fornecedor (codigo, nome, email) VALUES (null,  
“Pedro”, “pedro@gmail.com”);
```

```
INSERT INTO fornecedor (codigo, nome, email) VALUES (null, “Joaquina”,  
“jo_123@gmail.com”)
```

Método SELECT

O método SELECT simboliza a busca de verificação de informações das tabelas de nosso banco de dados, podendo atuar através do comando “SELECT” que serve para buscar registros da nossa tabela, a sintaxe é bem simples para esse processo:

```
“SELECT * FROM fornecedor;”
```

O resultado dessa sintaxe ocorre listando os registros que associamos no momento anterior, contribuindo com todas as informações destacadas.

Através do método SELECT podemos realizar a busca de uma tabela apenas por um atributo específico, que nesse caso será o e-mail, a sintaxe fica a seguinte:

```
“SELECT email FROM fornecedores;”
```

Essa busca irá acarretar na listagem de apenas os e-mails inseridos no nosso banco de dados e se por ventura houvesse a necessidade de listar apenas outro atributo seria necessário mudar a nomenclatura da sintaxe.

Método ORDER BY

O método ORDER BY serve para que possamos retornar nossos registros de forma ordenada, para tal podemos usar a seguinte consulta para ordenação:

```
“SELECT * FROM fornecedor ORDER BY nome asc;”
```

Através dessa sintaxe todos os registros da tabela fornecedor irão vir de forma ordenada pelo campo nome, utilizamos o termo “asc” para ordenar de forma ascendente, e o termo “desc” para ordenar de forma descendente.

Método UPDATE

Quando desejamos realizar a alteração de um registro devemos utilizar o comando “UPDATE” que serve da forma de possibilitar a edição dos campos da tabela atribuindo novos valores, conforme a sintaxe a seguir:

```
“UPDATE fornecedor SET nome=“Ricardo Arrigoni” WHERE codigo=1;”
```

Após esse comando podemos notar que o nome que inserimos em nosso banco foi alterado para “Ricardo Arrigoni”, onde podemos estar alterando outros atributos da mesma forma também.

Método DELETE

O método DELETE serve basicamente para realizar a exclusão de um registro no nosso banco, esse método é responsável por remover, apagar qualquer registro do banco de dados.

Para realizar a exclusão de um registro devemos utilizar a seguinte sintaxe:

`“DELETE FROM fornecedores WHERE codigo=3;”`

Essa sintaxe desempenha o papel de realizar a exclusão do código 3 do registro do nosso banco de dados.

MYSQL: contexto geral

O banco de dados MySql atua de forma direta com alguns benefícios diretos, conforme elencando anteriormente, podemos destacar alguns benefícios gerais, tais como a segurança avançada em que a arquitetura do SGBD atua, podemos destacar que esse banco atende diversas demandas de forma segura e garantindo as credenciais do usuário e do próprio empregador.

O SGBD oferece meios como o comando “grant” que libera permissões aos usuários e utilizadores, e o comando “revoke”, que cancela essas permissões impostas, através disso é possível atender as demandas de limitações que possam a ter com os usuários do sistema, entre tabelas e colunas do próprio banco de dados.

Outro fator crucial ao sistema no meio de serviços de backup envolvendo sua arquitetura de segurança envolve todo o processo de recuperação das informações, onde nelas inseridas podemos recuperar caso ocorra algum tipo de falha ou exclusão de forma errada.

O único fator que possivelmente seja uma desvantagem é que para aplicações de maior porte sejam indicados outros SGBD’s para gerenciamento, pois possuem um porte maior, contudo demonstrando os recursos avançados que suprem as necessidades solicitadas.

Considerações Finais

Concluimos este artigo da forma em que dedicamos essas informações sejam esclarecedoras para futuros aproveitamentos em relação ao desenvolvimento de qualquer tipo de banco de dados em formato MySql, onde possa ser retirado aproveitamento de conteúdo e possível contribuição com o andamento de qualquer desenvolvimento da área. Além de analisar tudo até aqui visto promover experiências práticas a partir desse documento.

Neste artigo esclarecemos possíveis dúvidas que ainda venham a ter, pois

possam ser descritas e identificadas no decorrer desse artigo, onde facilitem com o entendimento, melhorias e construção de um banco de dados do início até seus principais recursos e características.

Possibilitamos através desse artigo que outros profissionais tenham conhecimento de banco de dados MySQL e que possam fazer bom proveito para melhor esclarecimento de técnicas e meios de desenvolvimento e manipulação de registros do SGBD MySQL através desse artigo para fins de uso pessoal e corporativo.

Referências

MySQL: O que é, para que serve e suas vantagens. Ricardo, 2013. Disponível em:

<<https://linkoficial.com.br/2017/10/31/mysql-o-que-e-para-que-serve-e-suas-vantagens/>>. Acesso em: 14 de fevereiro de 2022.

Primeiros Passos no MySQL. linkoficial, 2022. Disponível em: <<https://www.devmedia.com.br/primeiros-passos-no-mysql/28438/>>. Acesso em: 25 de fevereiro de 2022.

MySQL Tutorial. Bruno, 2015. Disponível em: <<https://www.devmedia.com.br/mysql-tutorial/33309/>>. Acesso em: 25 de março de 2022.