



HACKING DAY

Bancos de Dados

MANIPULAÇÃO DE ARQUIVOS



Modos de acesso

Modo	Significado
r	Leitura. O Arquivo deve existir.
r+	Leitura e Escrita. O Arquivo deve existir.
w	Abre o arquivo somente para escrita no início do arquivo. Apagará o conteúdo do arquivo se ele já existir, criará um arquivo novo se não existir.
w+	Abre o arquivo para escrita e leitura, apagando o conteúdo pré-existente.
a	Abre o arquivo para escrita no final do arquivo. Não apaga o conteúdo pré-existente
a+	Abre o arquivo para escrita no final do arquivo e leitura.

Abrir e fechar arquivo

```
FILE *fp;  
fp = fopen ("saida.txt", "w");  
  
// Manipulação do arquivo  
fclose(fp);
```

Ler arquivo

```
int num1, num2, num3;
FILE *fp;

fp = fopen ("arquivo.txt", "r");
fscanf(fp, "%d %d %d", &num1, &num2, &num3);
printf("O arquivo contem os numeros %d, %d e %d!\n", num1, num2, num3);

fclose(fp);
```

Escrever em arquivo

```
FILE *fp;  
fp = fopen ("saida.txt", "w");  
fprintf(fp, "Introcomp 2017!\n");  
fclose(fp);
```

Escrever em arquivo

```
int num;
FILE *fp;

fp = fopen ("saida.txt", "w");
printf("Digite um numero: ");
scanf("%d", &num);
fprintf(fp, "Você digitou o número %d!\n", num);

fclose(fp);
```

COMMA-SEPARATED VALUES (CSV)



Comma-Separated Values (CSV)

- Permite a representação de dados tabulares em formato de texto puro
- Cada linha de texto representa uma linha da tabela
- O valor das células de uma linha são separadas por vírgula
- Especificação:
 - *Definido em texto puro*
 - *Um registro por linha do arquivo*
 - *Os campos de um registro são separados por vírgula*
 - *Cada registro possui a mesma sequência de campos*

Comma-Separated Values (CSV)

Ano	Marca	Modelo	Descrição	Preço
1997	Ford	E350	ac, abs, moon	3000.00
1999	Chevy	Venture "Extended Edition"		4900.00
1999	Chevy	Venture "Extended Edition, Very Large"		5000.00
1996	Jeep	Grand Cherokee	MUST SELL! air, moon roof, loaded	4799.00

Comma-Separated Values (CSV)

```
Ano,Marca,Modelo,Descrição,Preço
1997,Ford,E350,"ac, abs, moon",3000.00
1999,Chevy,"Venture ""Extended Edition""",,,4900.00
1999,Chevy,"Venture ""Extended Edition, Very Large""",,5000.00
1996,Jeep,Grand Cherokee,"MUST SELL!air, moon roof, loaded",4799.00
```

HANDS ON



Hands On

- Crie um laço de repetição que execute 5 vezes
 - Em cada iteração, peça ao usuário seu nome, idade, email e telefone
 - Para cada dado, escreva-o no arquivo e insira uma vírgula (caso você esteja lendo o telefone, a vírgula não deve ser adicionada)
 - Ao final de cada iteração, pule uma linha
-
- **DESAFIO:** Faça com que o programa peça as informações do usuário até que ele digite “0” (zero) como seu nome.

BANCOS DE DADOS



Bancos de Dados

- Armazenam dados de forma estruturada por meio de tabelas
- São de vital importância para empresas
- Tornaram-se a principal peça dos sistemas de informação.
- Permitem que os usuários busquem, insiram, modifiquem ou deletem dados de forma fácil e rápida
- Relacionamento via SGBD (Sistema de Gestão de Banco de Dados) - usaremos o MySQL

Bancos de Dados

- Tipos mais comuns de dados:
 - **INT**: armazena um valor inteiro
 - **FLOAT**: armazena um valor do tipo float
 - **BOOLEAN**: armazena TRUE ou FALSE
 - **VARCHAR**: armazena um texto de tamanho variável de até 255 caracteres
 - **TEXT**: armazena um texto de tamanho variável de até 65535 caracteres

Bancos de Dados

- **Chave primária:** Atributo ou combinação de atributos que possuem a propriedade de identificar de forma única uma linha da tabela.
- **Chave estrangeira:** ocorre quando um atributo de uma tabela for chave primária em outra relação.

Bancos de Dados

- Comandos SQL mais utilizados:
 - **SELECT**: seleciona dados da tabela.
 - **INSERT**: insere uma nova linha de dados na tabela
 - **UPDATE**: altera os dados de uma linha da tabela
 - **DELETE**: deleta uma linha da tabela
 - **DROP**: deleta uma tabela
 - **TRUNCATE**: limpa uma tabela

Bancos de Dados - SELECT

```
SELECT coluna1 FROM tabela;
```

```
SELECT coluna1, coluna2, coluna3 FROM tabela;
```

```
SELECT * FROM tabela;
```

Bancos de Dados - INSERT

```
INSERT INTO tabela (coluna) VALUES (valor);
```

```
INSERT INTO tabela (coluna1, coluna2) VALUES (valor1, valor2);
```

Bancos de Dados - UPDATE

```
UPDATE tabela SET coluna=valor WHERE uma_coluna=certo_valor;
```

```
UPDATE tabela SET coluna1=valor1, coluna2=valor2, coluna3=valor3 WHERE  
uma_coluna=certo_valor;
```

Bancos de Dados – DELETE, DROP e TRUNCATE

```
DELETE FROM tabela WHERE uma_coluna=tal_valor;
```

```
DROP TABLE tabela;
```

```
TRUNCATE TABLE tabela;
```

HANDS ON

