**AULA 15 – Chaves Estrangeiras e JOIN**

* Engines que suportam MySQL com suporte à Chaves Estrangeiras:
  + MyISAM
  + InnoDB
  + XtraDB
  + Além disso, as engines InnoDB e XtraDB suporta o que chamamos de ACID:
    - **Atomicidade:** toda transação deve ser atômica (ou toda a transação acontece ou nenhuma transação acontece);
    - **Consistência:** toda transação deve levar o banco de dados de um estado consistente a outro estado consistente;
    - **Isolamento:** quando acontecem 2 transações ao mesmo tempo, uma não pode interferir na outra;
    - **Durabilidade:** todo dado que foi inserido, manipulado etc. deve permanecer durável (de forma intacta, enquanto quiser que ele esteja lá);

* Para adicionar uma Chave Estrangeira no banco de dados, se usam os comandos:
  + ALTER TABLE <n\_tbl1>

ADD COLUMN <foreign\_key\_n\_cmp1><tipo\_cmp1><tam\_cmp1>;

(O parâmetro <tam\_cmp1> é opcional, dependendo das características da Chave Estrangeira)

* + ALTER TABLE <n\_tbl1>

ADD FOREIGN KEY (<foreign\_key\_n\_cmp1\_tbl2>)

REFERENCES <n\_tbl2>(<primary\_key\_n\_cmp1\_tbl2>);

(O parâmetro <primary\_key\_n\_cmp1\_tbl2> de <n\_tbl2> é o parâmetro <foreign\_key\_n\_cmp1\_tbl2> de <n\_tbl1>)

* Para inserir dados da chave primária de uma tabela como chave estrangeira em registros existentes de outra tabela, se usam os comandos:
  + UPDATE <n\_tbl1>

SET <foreign\_key\_n\_cmp1\_tbl2> = <v\_fk\_cmp1\_tbl2>

WHERE <primary\_key\_cmp1\_tbl1> = <v\_pk\_cmp1\_tbl1>;

* Para unir os dados de campos de uma tabela com os dados de campos de outra tabela e os visualizar, se usa o comando:
  + SELECT <n\_tbl1>.<n\_cmp1\_tbl1>, <n\_tbl1>.<n\_cmp2\_tbl1>, ..., <n\_tbl1>.<n\_cmpN\_tbl1>

FROM <n\_tbl1> JOIN <n\_tbl2>

ON <n\_tbl2>.<primary\_key\_cmp1\_tbl2> = <n\_tbl1>.<primary\_key\_cmp1\_tbl2>;

* + - É possível “apelidar” os campos com o comando AS:

[...] <nome> AS <apelido> [...]

* + O comando **JOIN** (ou **INNER JOIN**) liga e exibe somente os registros com relações entre as 2 tabelas.
  + O comando **OUTER JOIN** liga e exibe todos os registros com e sem relações de uma das tabelas.
    - Se for para exibir a partir da tabela à **ESQUERDA** do comando **OUTER JOIN**, se usa **LEFT OUTER JOIN** (ou **LEFT JOIN**).
    - Se for para exibir a partir da tabela à **DIREITA** do comando **OUTER JOIN** se usa **RIGHT OUTER JOIN** (ou **RIGHT JOIN**).