

Salvando alterações no banco de dados

Depois que preenchemos o formulário com os dados do aluno, o usuário vai poder alterar esses dados e quando ele clicar no botão de salvar, devemos decidir se vamos incluir um novo aluno no banco ou se vamos simplesmente alterar um aluno pré-existente. Para tomar essa decisão, vamos verificar se o aluno que vem do formulário já possui um **Id**. Se ele já possui **Id**, então isso significa que estamos editando um aluno que já estava no banco, caso contrário, estamos trabalhando com um novo aluno.

Comece então adicionando essa verificação no método `onOptionsItemSelected(...)` da classe `FormularioActivity` como abaixo:

```
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case R.id.menu_formulario_ok:
            Aluno aluno = helper.pegaAluno();
            AlunoDAO dao = new AlunoDAO(this);

            if (aluno.getId() == null) {
                dao.insere(aluno);
            } else {
                dao.altera(aluno);
            }

            dao.close();

            Toast.makeText(FormularioActivity.this, "Aluno " + aluno.getNome() + " salvo!", Toa:

            finish();
            break;
    }

    return super.onOptionsItemSelected(item);
}
```

Agora vamos implementar o método `altera(...)` no DAO. Neste método, você vai precisar pegar uma referência em modo de escrita para o banco de dados e então invocar o método `update(...)` nesse banco para atualizar os dados do aluno. Esse método irá exigir os seguintes dados como parâmetro:

- o nome da tabela
- os novos dados do aluno (na forma de um objeto `ContentValues`)
- a cláusula *where*, assim como fizemos no método `delete(...)`
- os parâmetros da cláusula *where* (um *array* de `String` contendo o **Id** do aluno a ser alterado)

No momento de criar o objeto `ContentValues`, lembre-se que já fizemos isso uma vez no método `insere(...)` do DAO. Extraia o trecho de código que cria um `ContentValues` para um método chamado `pegaDadosDoAluno(...)` como abaixo:

```
public ContentValues pegaDadosDoAluno(Aluno aluno) {
    ContentValues dados = new ContentValues();
    dados.put("nome", aluno.getNome());
    dados.put("endereco", aluno.getEndereco());
    dados.put("telefone", aluno.getTelefone());
    dados.put("site", aluno.getSite());
    dados.put("nota", aluno.getNota());
    return dados;
}
```

Teste seu aplicativo novamente e verifique se agora você consegue atualizar os dados dos alunos!