02

Praticando com consultas mais complicadas

Algumas consultas são mais difíceis de serem testadas, simplesmente porque seu cenário é mais complicado. Nesses casos, precisamos facilitar a criação de cenários.

Veja, por exemplo, o método `porPeriodo(Calendar inicio, Calendar fim)`, do nosso `LeilaoDao`. Ele devolve todos os leilões que foram criados dentro de um período e que ainda não foram encerrados:

```
public List<Leilao> porPeriodo(Calendar inicio, Calendar fim) {
    return session.createQuery("from Leilao l where l.dataAbertura " +
            "between :inicio and :fim and l.encerrado = false")
            .setParameter("inicio", inicio)
            .setParameter("fim", fim)
            .list();
}
```

Para testarmos esse método, precisamos pensar em alguns cenários, como: - Leilões não encerrados com data dentro do intervalo devem aparecer - Leilões encerrados com data dentro do intervalo não devem aparecer - Leilões encerrados com data fora do intervalo não devem aparecer - Leilões não encerrados com data fora do intervalo não devem aparecer

Vamos começar pelo primeiro cenário. Vamos criar 2 leilões não encerrados, um com data dentro do intervalo, outro com data fora do intervalo, e garantir que só o primeiro estará lá dentro. O método é grande, mas está comentado:

```
@Test
public void deveTrazerLeiloesNaoEncerradosNoPeriodo() {

    // criando as datas
    Calendar comecoDoIntervalo = Calendar.getInstance();
    comecoDoIntervalo.add(Calendar.DAY_OF_MONTH, -10);
    Calendar fimDoIntervalo = Calendar.getInstance();
    Calendar dataDoLeilao1 = Calendar.getInstance();
    dataDoLeilao1.add(Calendar.DAY_OF_MONTH, -2);
    Calendar dataDoLeilao2 = Calendar.getInstance();
    dataDoLeilao2.add(Calendar.DAY_OF_MONTH, -20);

    Usuario mauricio = new Usuario("Mauricio Aniche",
            "mauricio@aniche.com.br");

    // criando os leiloes, cada um com uma data
    Leilao leilao1 =
            new Leilao("XBox", 700.0, mauricio, false);
    leilao1.setDataAbertura(dataDoLeilao1);
    Leilao leilao2 =
            new Leilao("Geladeira", 1700.0, mauricio, false);
    leilao2.setDataAbertura(dataDoLeilao2);

    // persistindo os objetos no banco
    usuarioDao.salvar(mauricio);
    leilaoDao.salvar(leilao1);
    leilaoDao.salvar(leilao2);
```

```
        // invocando o metodo para testar
        List<Leilao> leiloes =
                leilaoDao.porPeriodo(comecoDoIntervalo, fimDoIntervalo);

        // garantindo que a query funcionou
        assertEquals(1, leiloes.size());
        assertEquals("XBox", leiloes.get(0).getNome());
    }
```

Ele passa. Vamos ao próximo cenário: leilões encerrados devem ser ignorados pela consulta. Nesse caso, criaremos apenas um leilão encerrado, dentro do intervalo. Esperaremos que a query não devolva nada:

```
    @Test
    public void naoDeveTrazerLeiloesEncerradosNoPeriodo() {

        // criando as datas
        Calendar comecoDoIntervalo = Calendar.getInstance();
        comecoDoIntervalo.add(Calendar.DAY_OF_MONTH, -10);
        Calendar fimDoIntervalo = Calendar.getInstance();
        Calendar dataDoLeilao1 = Calendar.getInstance();
        dataDoLeilao1.add(Calendar.DAY_OF_MONTH, -2);

        Usuario mauricio = new Usuario("Mauricio Aniche",
                "mauricio@aniche.com.br");

        // criando os leiloes, cada um com uma data
        Leilao leilao1 =
                new Leilao("XBox", 700.0, mauricio, false);
        leilao1.setDataAbertura(dataDoLeilao1);
        leilao1.encerra();

        // persistindo os objetos no banco
        usuarioDao.salvar(mauricio);
        leilaoDao.salvar(leilao1);

        // invocando o metodo para testar
        List<Leilao> leiloes =
                leilaoDao.porPeriodo(comecoDoIntervalo, fimDoIntervalo);

        // garantindo que a query funcionou
        assertEquals(0, leiloes.size());
    }
```

Montar cenários é o grande segredo de um teste de integração. Queries mais complexas exigirão cenários de teste mais complexos. Nos cursos anteriores, estudamos sobre como melhorar a escrita dos testes, como Test Data Builders, e etc. Você pode (e deve) fazer uso deles para facilitar a escrita dos cenários!