

Spring Professional Exam Tutorial v5.0

Question 04

Question 04 - When and where do you use @Transactional in testing?

When - you use `@Transactional` annotation in testing whenever you want to run some part of the code that can alter state of transactional resource, for example database. Usage of this annotation allows you to mark code that should execute under transaction and allows to rollback all changes made by test, allowing other tests to pick from clear state.

By default, transaction will be rolled-back for each test which was executed with `@Transactional` annotation.

You can gain additional control over transaction when using `@Transactional` annotation with:

- ▶ `@Rollback` - explicitly state to rollback transaction
- ▶ `@Commit` - indicates that after test execution, transaction should be committed
- ▶ `@BeforeTransaction` - indicates method that should be executed before transaction is started
- ▶ `@AfterTransaction` - indicates method that should be executed after transaction is started

Question 04 - When and where do you use @Transactional in testing?

Where - @Transactional can be used:

- ▶ On top of the class - each test method in class will be executed in transaction

```
@RunWith(SpringRunner.class)
@ContextConfiguration(classes = ApplicationConfiguration.class)
@Transactional
public class ApplicationServiceIntegrationTest {

    ...

}
```

- ▶ On top of the method - test method will be executed in transaction

```
@RunWith(SpringRunner.class)
@ContextConfiguration(classes = ApplicationConfiguration.class)
public class ApplicationServiceIntegrationTest {

    @Transactional
    @Test
    public void shouldRegisterGuests() {

        ...

    }

}
```

