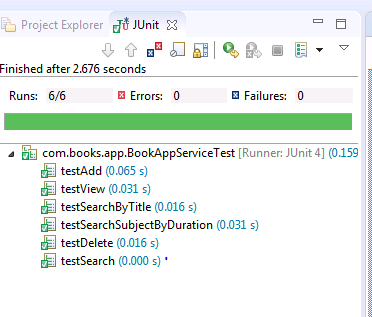
**SPRING TEST ASSIGNMENT 1**

**Results**



**BookAppServiceTest.java**

**package** com.books.app;

**import** **static** org.junit.Assert.*assertEquals*;

**import** **static** org.mockito.Mockito.*times*;

**import** **static** org.mockito.Mockito.*verify*;

**import** **static** org.mockito.Mockito.*when*;

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.Optional;

**import** org.junit.Before;

**import** org.junit.Test;

**import** org.junit.runner.RunWith;

**import** org.mockito.InjectMocks;

**import** org.mockito.Mock;

**import** org.mockito.MockitoAnnotations;

**import** org.springframework.test.context.junit4.SpringRunner;

**import** com.books.app.dao.IBookRepository;

**import** com.books.app.dao.ISubjectRepository;

**import** com.books.app.exception.BookAppException;

**import** com.books.app.model.Book;

**import** com.books.app.model.Subject;

**import** com.books.app.service.BookAppServiceImpl;

@RunWith(SpringRunner.**class**)

**public** **class** BookAppServiceTest {

@Mock

**private** ISubjectRepository subjRepo;

@Mock

**private** IBookRepository bookRepo;

@InjectMocks

**private** BookAppServiceImpl bookAppService;

@Before

**public** **void** setup(){

MockitoAnnotations.*initMocks*(**this**);

}

@Test

**public** **void** testView() **throws** BookAppException {

List<Book> bookList = **new** ArrayList<Book>();

Subject sub1 = **new** Subject();

sub1.setSubtitle("Subject 1");

Book book1 = **new** Book();

book1.setBookId(1);

book1.setTitle("Book 1");

book1.setPrice(10.00);

book1.setVolume(1);

book1.setSubject(sub1);

bookList.add(book1);

Subject sub2 = **new** Subject();

sub1.setSubtitle("Subject 2");

Book book2 = **new** Book();

book1.setBookId(2);

book1.setTitle("Book 2");

book1.setPrice(20.00);

book1.setVolume(2);

book1.setSubject(sub2);

bookList.add(book2);

*when*(bookRepo.findAll()).thenReturn(bookList);

List<Book> result = bookAppService.view();

*assertEquals*(2, result.size());

}

@Test

**public** **void** testSearch() **throws** BookAppException {

Subject sub1 = **new** Subject();

sub1.setSubtitle("Subject 1");

Book book1 = **new** Book();

book1.setBookId(1);

book1.setTitle("Book 1");

book1.setPrice(10.00);

book1.setVolume(1);

book1.setSubject(sub1);

*when*(bookRepo.findById(1)).thenReturn(Optional.*of*(book1));

Book result = bookAppService.search(1);

*assertEquals*(1, result.getBookId());

*assertEquals*("Book 1", result.getTitle());

*assertEquals*(1, result.getVolume());

}

@Test

**public** **void** testSearchByTitle() **throws** BookAppException {

List<Book> bookList = **new** ArrayList<Book>();

Subject sub1 = **new** Subject();

sub1.setSubjectId(1);

sub1.setSubtitle("Subject 1");

sub1.setDurationInHours(20);

Book book1 = **new** Book();

book1.setBookId(1);

book1.setTitle("Book 1");

book1.setPrice(10.00);

book1.setVolume(1);

book1.setSubject(sub1);

bookList.add(book1);

*when*(bookRepo.findAll()).thenReturn(bookList);

*when*(bookRepo.findById(1)).thenReturn(Optional.*of*(book1));

Book result = bookAppService.searchByTitle("Book 1");

*assertEquals*(1, result.getBookId());

*assertEquals*("Book 1", result.getTitle());

*assertEquals*(1, result.getVolume());

}

@Test

**public** **void** testSearchSubjectByDuration() **throws** BookAppException {

List<Subject> subjectList = **new** ArrayList<Subject>();

Subject sub1 = **new** Subject();

sub1.setSubjectId(1);

sub1.setSubtitle("Subject 1");

sub1.setDurationInHours(20);

subjectList.add(sub1);

*when*(subjRepo.findAll()).thenReturn(subjectList);

*when*(subjRepo.findById("Subject 1")).thenReturn(Optional.*of*(sub1));

List<Subject> result = bookAppService.searchSubjectByDuration(20);

*assertEquals*(1, result.size());

}

@Test

**public** **void** testAdd() **throws** BookAppException {

List<Subject> subjectList = **new** ArrayList<Subject>();

Subject sub1 = **new** Subject();

sub1.setSubtitle("Subject 1");

Book book1 = **new** Book();

book1.setBookId(1);

book1.setTitle("Book 1");

book1.setPrice(10.00);

book1.setVolume(1);

book1.setSubject(sub1);

subjectList.add(sub1);

*when*(subjRepo.findAll()).thenReturn(subjectList);

*when*(bookRepo.save(book1)).thenReturn(book1);

String result = bookAppService.add(book1, book1.getSubject().getSubtitle());

*assertEquals*("Book 1", result);

}

@Test

**public** **void** testDelete() **throws** BookAppException{

Book book1 = **new** Book();

book1.setBookId(1);

book1.setTitle("Book 1");

book1.setPrice(10.00);

book1.setVolume(1);

bookAppService.delete(1);

*verify*(bookRepo, *times*(1)).deleteById(1);

}

}