

Spring MockMvc ve Birim Testler



MockMvc ve Birim Testler

- MockMvc ile normal JUnit testleri içerisinde **Controller** sınıflarının yaratılıp herhangi **arka katmandan bağımsız** birim teste tabi tutulması da sağlanabilir
- Controller nesnenin ihtiyaç duyduğu herhangi bir **servis nesnesi vs mock olarak** kendisine sağlanır

MockMvc ve Birim Testler

```
@RunWith(MockitoJUnitRunner.class)
public class StandaloneMvcTests {
```

```
    private MockMvc mockMvc;
```

```
    @Mock
    private PetClinicService petClinicService;
```

```
    @Before
```

```
    public void setUp() {
```

```
        PetClinicController petClinicController = new PetClinicController();
        petClinicController.setPetClinicService(petClinicService);
```


```
        Mockito.when(petClinicService.getVets()).thenReturn(
            Arrays.asList(new Vet());
        );
```

```
        mockMvc = MockMvcBuilders.standaloneSetup(petClinicController).build();
```

```
    }
```

```
}
```

Test edilecek Controller sınıfının
bağımlılıkları mock'lanır



MockMvc ve Birim Testler

```
@Test
public void testGetVets() throws Exception {
    mockMvc.perform(MockMvcRequestBuilders.get("/vets"))
        .andExpect(MockMvcResultMatchers.print());
}

@Test
public void testFindOwners() throws Exception {
    mockMvc.perform(
        MockMvcRequestBuilders.get("/owners").param("lastName", "Davis"))
        .andExpect(MockMvcResultMatchers.view().name("ownersList"))
        .andExpect(MockMvcResultMatchers.model().attributeExists("owners"));
}
```

İletişim

- **Harezmi** Bilişim Çözümleri
- Kurumsal Java Eğitimleri
- <http://www.java-egitimleri.com>
- info@java-egitimleri.com

