

ASP.NET Core MVC

ViewBag & ViewData

Ingo Köster

Diplom Informatiker (FH)

Datenaustausch

- › Für den Datenaustausch zwischen Controller und View stehen mehrere Objekte zur Verfügung
 - › ViewBag
 - › ViewData
- › Diese bieten die Möglichkeit neben einem Model weitere Daten an eine View zu übertragen
- › Dienen zum Austausch von kleineren Datenmengen zwischen Controller und View

Geeignete Daten

- › ViewBag und ViewData eignen sich eher für kleinere Datenmengen wie z.B.
 - › DropDown Listen
 - › Einkaufswagen
 - › Fehlermeldungen
- › Mittels ViewBag und ViewData können Daten nur vom Controller zur View transportiert werden (unidirektional)

ViewBag

- › ViewBag ist ein dynamisches Objekt
 - › Datentyp `dynamic`
- › In ein dynamisches Objekt können beliebige Daten geschrieben werden
- › Ein ViewBag Objekt hat keine definierten Eigenschaften bis diese angehängt wurden
- › Ein ViewBag Objekt unterstützt daher kein Intellisense

Im Controller

- › Das ViewBag Objekt wird von der Basisklasse Controller vererbt und kann um beliebige Eigenschaften erweitert werden

```
public IActionResult ViewBagBeispiel()
{
    ViewBag.MeinText = "Hallo Welt";
    ViewBag.DieAnzahl = 5;
    return View();
}
```

In der View

- › In der View wird über das ViewBag Objekt mittels einer Razor Anweisung auf die Daten zugegriffen

```
<ul>
```

```
    @for (int i = 0; i < ViewBag.DieAnzahl; i++)
```

```
    {
```

```
        <li>@ViewBag.MeinText</li>
```

```
    }
```

```
</ul>
```

ViewData

ViewData

- › Ein Dictionary, welches Daten enthält, die zwischen dem Controller und der View übergeben werden sollen
- › Ist von der Klasse ViewDataDictionary abgeleitet
- › Verwendet Key/Value-Paare

Im Controller

```
public IActionResult ViewDataBeispiel() {  
    List<string> zoo = new List<string>();  
    zoo.Add("Hund");  
    zoo.Add("Katze");  
    zoo.Add("Maus");  
    ViewData["Haustiere"] = new SelectList(zoo);  
    return View();  
}
```

In der View

- › Zugriff über das Property `@ViewData[""]`, in den Klammern wird der Key angegeben

```
ViewData["Name"] = "Tom";           // Schreiben  
@ViewData["Name"]                  // Lesen
```

- › Mittels des HTML Helper für DropDown Listen (ohne das For) kann auf die ViewData Inhalte zugegriffen werden

```
<div>  
    @Html.DropDownList("Haustiere")  
</div>
```

ViewBag vs. ViewData

```
public IActionResult Index() {  
    ViewBag.Name = "Tom";  
    return View();  
}
```

@ViewBag.Name

```
public IActionResult Index() {  
    ViewData["Name"] = "Tom";  
    return View();  
}
```

@ViewData["Name"]

Hinweise

- › Neben ViewBag und ViewData stehen weitere Objekte zur Verfügung
- › Zur Übertragung zwischen View zu View
 - › Das Objekt TempData
- › Zum Speichern von Informationen in einer Session
 - › Das Objekt Session