

Index

A

AddCharts extension method, 235–236

AddToBasket method, 37

ASP.NET Core

 pizza service

 add, new controller, 132

 API Controller, 132

 empty PizzasController, 133

 GetPizzas method, 134

 JSON-encoded list, 135

 modifications,

 PizzasController, 133–134

 naming, controller, 133

 PizzaPlace.Server project, 130, 135

 results, 135

 Startup Class' Configure

 method, 131–132

 UseBlazor method, 132

 services and single

 responsibility, 130

AutoIncrement method, 27, 28

B

BankRepository, 112, 113

Blazor component

 Alert component, 55–58

 component library project

 creation, 67

 CSHTML files, 71

 deletion, project, 69

 DismissableAlert.cshtml, 69, 70

 MyFirstBlazor.Client.csproj file, 71

 tag helpers, 72

 Timer.cs file, 68

 Visual Studio, 70

CSHTML file, 53

lifecycle

 IDisposable, 88–89

 OnInit and OnInitAsync

 methods, 86–87

 OnParametersSet and

 OnParametersSetAsync

 methods, 87

MyFirstBlazor project, 53

PizzaItem.g.cs Generated File, 97–98

pizzas

 creation, 73–75

 CustomerEntry component, 81–86

 ShoppingBasket component, 76–78

 update, state object, 75–76

 ValidationError component, 78–80

Razor, 99

SurveyPrompt component, 54–55

templated components

 context argument, 93

 Context parameters, 93

 Fetch data link, 93

 FetchData component, 91–93

 grid, 89–91

 Razor templates, 94–96

 type parameter, 94

INDEX

Blazor component (*cont.*)

- view and view-model
 - child component, 61
 - component-to-component data binding, 64–66
 - DismissableAlert component, 59–61
 - timer component, 62–64
- Visual Studio, 55–56

Blazor component library

- Chart.js, 226–229
- ChartOptions class, 230–231
- creation, 223–224
- LineChartData class, 232–233
- project, addition, 224–226

Blazor Language Services, 3

Blazor layout components, 188–190

Blazor project's Startup

- class, 109, 173

Blazor services

- configuring dependency injection, 118, 119
- GetMenu method returns, 116, 117
- HardCodedMenuService class, 117
- IMenuService interface, 116
- Index Component, 115, 116
- Index Component's Menu, 118
- OnInitAsync method, 117
- ordering pizzas
 - asynchronous PlaceOrder Method, 121
 - asynchronous lambda function, 122
- ConsoleOrderService, 120
- dependency injection, configuration, 121, 122
- IOrderService, 119–120
- Order button, 122

- PlaceOrder method, 120
 - Task.CompletedTask, 120
 - State.Menu property, 116, 118
- ### Bootstrap process, 16–17

C

C#

- ElementRef Property, 218–219
 - Glue function, 213
 - Interop, 214–217
 - JSRuntime, 214
- ### Catchall route template, 200
- ### Chrome browser
- array, WeatherForecast instances, 167, 168
 - debugger, 166
 - feedback with a slow network, 167
 - OnInitAsync method, 167
- ### Client Blazor Project
- App component, 14
 - Configure method, 14
 - Index component, 14
 - index.html, 13
 - layout components, 15–16
- ### Communication, microservices
- HttpClient Class (*see* HttpClient Class)
 - retrieving data
 - Blazor Project's Startup Class, 173
 - ConfigureServices method, 174
 - loading UI, PizzaList Component, 176, 177
 - MenuService class, 174
 - PizzaPlace app, 176
 - progress bar, 177
 - replace HardCodedMenuService, MenuService, 175

- storing changes
 - database update,
 - orders, 178–183
- Counter page, 22

D

- Data binding
 - Counter.cshtml, 27, 28
 - definition, 19
 - SurveyPrompt.cshtml, 19, 20
- Data storage and microservices
 - Entity Framework Core (*see* Entity Framework Core)
 - code first migrations, 141–142
 - database server’s connection string, 143, 144
 - functionality, Pizza microservice (*see* Pizza microservice)
 - Postman, 151–159
 - REST (*see* Representational State Transfer (REST))
- Dependency injection, configuration
 - disposing, 114
 - IoCC, 109
 - options, 109, 110
 - scoped, 111–114
 - serviceProvider, 109
 - Singleton, 110
 - Startup Class, 109
- Dependency inversion
 - Blazor PizzaPlace app, 101
 - component, ProductsService, 102
 - dependency injection, 105, 106
 - IoCC (*see* Inversion-of-Control Container (IoCC))
 - Principle, 103–105
 - ProductList component, 102

- service object, 101
- traditional layered approach, 102
- Disposing dependencies, 114
- Document Object Model (DOM), 19
- Dynamic pages, 187

E

- Entity Framework Core
 - code first approach
 - add, NuGet packages, 137
 - modelBuilder, 139
 - modification, Pizza Class, 138–139
 - NuGet, 137
 - OnModelCreating method, 139
 - Pizza class, 136
 - PizzaPlaceDbContext
 - Class, 138–139
 - code first migrations
 - application configuration file, 141
 - appsettings.json, 142
 - ASP.NET Core, 141
 - database server’s connection string, 142–144
 - dependencies, 141
 - IServiceCollection, 140
 - Startup Class’ Constructor, 140–141
 - Startup.ConfigureServices
 - Method, 140
 - create migration
 - CreatingPizzaDb.cs File, 145–146
 - dotnet command, 145
 - generate database
 - connection, SQL Operations Studio, 149
 - SQL Server Object Explorer, PizzaDb Database, 148
 - Tool’s Output, 147

INDEX

Entity Framework Core (*cont.*)
 Package Manager Console, 144
 result, adding first migration, 145
 tool, 147
Event arguments, 23
Event binding syntax, 23

F

FetchData Component, 164, 165
FetchData Razor file, 167

G

GetJsonAsync extension method, 169–171
GetPizza Method, 41, 134
Grid templated component, 91

H

Hamburger button, 196
Hard-Coded ProductService, 104
HasErrors property, 46
HttpClient Class
 browser, invoking service, 163
 Client Project, 164–166, 168
 HttpClientJsonExtensions class
 (*see* HttpClientJsonExtensions
 methods)
 MyFirstBlazor.Server project, 161
 MyFirstBlazor.Shared project, 163, 164
 SampleDataController class, 161, 162
 WeatherForecasts method, 163
HttpClientJsonExtensions methods
 GetJsonAsync, 169–171
 PostJsonAsync, 171
 SendJsonAsync, 172, 173
HTTP Status Codes, 127

Hypertext Transfer Protocol (HTTP)
 CERN, 125
 electronic document format, 126
 HTTP headers, 127
 status code, 127
 URI and verbs, 126

I

IChartInterop interface, 234–235
IDisposable, 114
IMenuService, 117
IncrementCount() method, 23
Inject Attribute, property injection, 108
Installation, Blazor
 ASP.NET Core, 3
 .NET Core, 1
 Visual Studio 2017, 2, 3
 Visual Studio Code Extensions tab, 3, 4
 VS/Code templates, 4, 5
Integrated development
 environments (IDEs), 2
Inversion-of-Control
 Container (IoCC), 106
 properties, 107, 108
 ProductService's Constructor, 107
IProductsService interface, 103, 104

J, K

JavaScript Glue Class, 233

L

Lambda function, 24
Layout components, SPA
 @layout directive, 190–192
 simple nested layout, 192–194

LineChart component, 236–237
 index, 238–239
 JS Extension method, 239–240
 LINQ Select method, 41

M

MyFirstBlazor.Client project, 165, 166

N

.NET methods
 OnInit method, 221–222
 UpdateCounter method, 220–221
 watch Function, 220
 NuGet window, 137

O

OnAfterRender and OnAfterRenderAsync
 methods, 87–88
 One-Way Data Binding
 conditional attributes, 22, 23
 syntax, 21, 22
 OnInit method, 85

P, Q

Pizza microservice
 GetPizza method, 150
 InsertPizza method, 150, 151
 PizzaPlaceDbContext, 150
 Pizzas property, 150
 PizzasController class, 149, 150
 retrieve, 150
 PizzaPlace.Client project, 174
 PizzaPlaceDbContext
 Class, 138–139

PizzaPlace project
 OnInitAsync method, 117
 PlaceOrder method, 120
 PizzaPlace.Server project, 135
 PizzaPlace.Shared project, 116
 Pizza Place Single Page application
 Basket class, 33
 creation, 29, 30
 Customer class, 32
 customer details, 42–44
 customer information, validating
 Any Extension Method, 46
 disable checkout button, 51, 52
 GetErrors method, 46
 INotifyDataErrorInfo, 46–48
 validation error, 48–51
 DebuggingExtensions class, 44, 45
 displaying menu, 34, 36
 Menu class, 31
 Pizza class, 31
 shopping basket
 Basket class, 41
 displaying, 39, 40
 image folder conversion, 39
 package installation, 39
 pizza ordering, 37
 Pizza Place menu, 38
 state class, 38
 tuples, 41
 Spiciness class, 31
 State class, 33, 34
 UI class, 33
 Pizzas and orders
 compiler error(s), 182
 Customer class, 179, 180
 HasForeignKey<Order> method, 182
 mapping, PizzaOrder class, 178
 migration, database, 183

INDEX

Pizzas and orders (*cont.*)

- Order Class, 178
- OrdersController Class, 183, 184
- OrderService Class, 185, 186
- PizzaOrder Class, 178
- PizzaOrders property, 180, 181
- PizzaPlaceDbContext class, 181, 182
- primary key Id, 182
- PizzasController, 133
- PlaceOrder method, 43
- PostJsonAsync Method's Signature, 171
- Postman
 - installation, 151
 - REST Calls
 - add Content-Type header, 157
 - adding headers, 155
 - empty list of pizzas, 156
 - GET request, 155
 - get started, 153
 - pizza object, JSON, 157
 - POST response, 158
 - POST request, 156
 - Saving, 154
 - storage, list of pizzas, 159
 - web page, 152
- Principle, Dependency Inversion, 103–105
- ProductList component, 104, 105
- ProductsService, 102, 105, 108
- Property dependency injection, 107, 108
- PutJsonAsync Method's Signature, 172

R

- RemoveAt method, 41
- Representational State Transfer (REST)
 - building simple microservice
 - (*see* ASP.NET Core)
 - HTTP (*see* Hypertext Transfer Protocol (HTTP))

server functionality

- CRUD operations, 127
 - HTTP headers, 127
 - JSON, 128
 - POSTing, 129
 - retrieve, list of pizzas, 128
 - unique id, 129
- ### Routing, SPA
- constrains, 199
 - installation, 194
 - NavLink component, 197
 - NavMenu component, 195–196
 - parameters, 198

S

- Scoped dependencies, 111–114
- SendJsonAsync Method, 172, 173
- Server Project's Configure Method, 11
- Shared WeatherForecast Class, 12–13
- Single page application (SPA)
 - base tag, 202
 - description, 187, 188
 - navigating, anchor, 200
 - navigating, code, 200–201
 - NavLink component, 200
 - routing (*see* Routing, SPA)
- Singleton dependencies, 110
- Singleton pattern, 203
 - adding, PizzaInfo component, 211–212
 - callback, PizzaList component, 208–209
 - CurrentPizza property, 205–206
 - dependency injection, 203–204
 - display, pizza's information, 206–207
 - navigation, PizzaList
 - component, 209–211
- SpicinessImage method, 36
- SQL Server Object Explorer, 142
- StateHasChanged method, 28

Static page, 187
System.ValueTuple NuGet package, 39

T

ToggleAlert method, 58
TransferService, 112
Transient dependencies, 111
Two-Way Data Binding
 date format, 26
 syntax, 24–26

U

Universal Resource Identifiers (URI)
 and verbs, 126
UseBlazor method, 132

UseResponseCompression
 Middleware, 12
U2U.Components.Chart project, 229

V

Visual Studio generation
 dotnet cli generation, 7, 8
 project creation, 6, 7
 running
 counter page, 9
 Fetch data tab, 9, 10
 home page, 8

W, X, Y, Z

WeatherForecast class, 163