# Webes és mobil programozás F#-ban 7. gyakorlat

## Berezvai Dániel jegyzete <http://elte.3ice.hu/>

15 perc késés

15 perc szenvedés a WiFi-vel

15 perc szenvedés admin belépéssel

Végül tanári gépre pendrive

Csak én jöttem be.

## Bevezetés

Kedvenc nyelvem a Java.

Mennyi szenvedés egy webszerver megírása...

F# esetén a nyelvben van minden tool.

Database

JQuery erősen tipizált.

Metaprogramozás

Computation expressions

Kódblokk elé írt kulcsszó (pl. query) és a benne levő hint

Rövidebb, olvashatóbb kód.

## F# Interactive part 1

1+1;;

Starting F# Interactive...

val ( CloudSharper F# Interactive Runtime initializing... ) : unit = ()

val ( CloudSharper F# Interactive Runtime initialized. ) : unit = ()

val it : int = 2

<@

Button **[**Text "Click me!"**]**

**|>!** OnClick **(fun** **\_** **\_** **->** JavaScript**.**Alert "Clicked!"**)**

**@>**

**val** it **:** Quotations**.**Expr**<**Element**>** **=**

Charting**.**Combine **[**

Charting**.**Line **[**

"Paris"**,** 2.4

"Madrid"**,** 1.5

"Budapest"**,** 3.2

"London"**,** 2.1

**]**

**|>** Charting**.**WithLegend "First data point"

Charting**.**Column **[**

"Paris"**,** 1.7

"Madrid"**,** 0.2

"Budapest"**,** 0.8

"London"**,** 1.7

**]**

**|>** Charting**.**WithLegend "Second data point"

**]**

**|>** Charting**.**WithXTitle "City"

**val** it **:** Charting**.**Chart **=**

Charting**.**Combine **[**

Charting**.**Line **[**

"Paris"**,** 2.4

"Madrid"**,** 1.5

"Budapest"**,** 3.2

"London"**,** 2.1

**]**

**|>** Charting**.**WithLegend "First data point"

Charting**.**Column **[**

"Paris"**,** 1.7

"Madrid"**,** 0.2

"Budapest"**,** 0.8

"London"**,** 1.7

**]**

**|>** Charting**.**WithLegend "Second data point"

**]**

**|>** Charting**.**WithXTitle "City"

**val** it **:** Charting**.**Chart **=**

Charting**.**Combine **[**

Charting**.**Line **[**

"Paris"**,** 2.4

"Madrid"**,** 1.5

"Budapest"**,** 3.2

"London"**,** 2.1

**]**

**|>** Charting**.**WithLegend "First data point"

Charting**.**Column **[**

"Paris"**,** 1.7

"Madrid"**,** 0.2

"Budapest"**,** 0.8

"London"**,** 1.7

**]**

**|>** Charting**.**WithLegend "Second data point"

**]**

**|>** Charting**.**WithXTitle "City"

**val** it **:** Charting**.**Chart **=**

Charting**.**Combine **[**

Charting**.**Line **[**

"Paris"**,** 2.4

"Madrid"**,** 1.5

"Budapest"**,** 3.2

"London"**,** 2.1

**]**

**|>** Charting**.**WithLegend "First data point"

Charting**.**Column **[**

"Paris"**,** 1.7

"Madrid"**,** 0.6

"Budapest"**,** 0.8

"London"**,** 1.7

**]**

**|>** Charting**.**WithLegend "Second data point"

**]**

**|>** Charting**.**WithXTitle "City"

**val** it **:** Charting**.**Chart **=**

Charting**.**Column **[**

"Alice"**,** 3.5

"Bob"**,** 2.1

"Charles"**,** 4.1

"Dennis"**,** 2.5

**]**

**|>** Charting**.**WithLegend "Math" **//** Add a legend **with** the given caption**.**

**|>** Charting**.**WithXTitle "Student" **//** Set the horizontal axis title**.**

**|>** Charting**.**WithYTitle "GPA" **//** Set the vertical axis title**.**

**|>** Charting**.**WithYBounds 0. 5. **//** Set the vertical axis **to** go from 0 **to** 5.

**|>** Charting**.**WithTheme "Desert" **//** Change the data color theme**.**

**//** The list **of** themes is available at http**://**dojotoolkit**.**org**/**api

**//** under dojox**/**charting**/**themes**.**

**val** it **:** Charting**.**Chart **=**

Charting**.**Combine **[**

Charting**.**Column **[**

"Alice"**,** 3.5

"Bob"**,** 2.1

"Charles"**,** 4.1

"Dennis"**,** 2.5

**]**

**|>** Charting**.**WithLegend "Math"

Charting**.**Line **[**

"Alice"**,** 3.2

"Bob"**,** 1.5

"Charles"**,** 1.4

"Dennis"**,** 4.1

**]**

**|>** Charting**.**WithLegend "PE"

**]**

**|>** Charting**.**WithXTitle "Student"

**|>** Charting**.**WithYTitle "GPA"

**|>** Charting**.**WithYBounds 0. 5.

**val** it **:** Charting**.**Chart **=**

Charting**.**Line **[**

"Alice"**,** 3.2

"Bob"**,** 1.5

"Charles"**,** 1.4

"Dennis"**,** 4.1

**]**

**val** it **:** Charting**.**Chart **=**

Charting**.**Line **[**

1**,**1

2**,**4

3**,**9

4**,**16

**]**

**val** it **:** Charting**.**Chart **=**

Charting**.**Line **[**

**for** i **in** **[-**System**.**Math**.**PI **..** System**.**Math**.**PI**]** **->** i **,** i**\***i

**]**

**val** it **:** Charting**.**Chart **=**

Charting**.**Line **[**

**for** i **in** **[-**System**.**Math**.**PI **..** **.**1 **..** System**.**Math**.**PI**]** **->** i **,** i**\***i

**]**

Charting**.**Line **[**

**for** i **in** **[-**System**.**Math**.**PI **..** 0.1 **..** System**.**Math**.**PI**]** **->** i **,** i**\***i

**]**

**val** it **:** Charting**.**Chart **=**

**open** System

Charting**.**Line **[**

**for** i **in** **[-**System**.**Math**.**PI **..** 0.1 **..** System**.**Math**.**PI**]** **->** i **,** Math**.**Sin**(**i**\***i**)**

**]**

**val** it **:** Charting**.**Chart **=**

1

**val** it **:** int **=** 1

**<@**1**@>**

**val** it **:** Quotations**.**Expr**<**int**>** **=**

**<@** Div **[**Text "Text here"**]** **@>**

**val** it **:** Quotations**.**Expr**<**Element**>** **=**

**<@** Text "Text here" **@>**

**val** it **:** Quotations**.**Expr**<**IPagelet**>** **=**

Div **[**Attr**.**Class "adam"**]** **-<** Div **[**Text "Text here"**]**

**<@** Div **[**Attr**.**Class "adam"**]** **-<** Div **[**Text "Text here"**]** **]** **@>**

**<@** Div **[**Attr**.**Class "adam"**]** **-<** Div **[**Text "Text here"**]** **]** **@>;;**

**-----------------------------------------------------^**

stdin**(**167**,**54**):** error FS0010**:** Unexpected symbol ']' **in** quotation literal**.** Expected **end** **of** quotation **or** other token**.**

**<@** Button **[**Text "Click me!"**]**

**|>!** OnClick **(fun** e args **->** e**.**Text **<-** "Hello!"**)** **@>**

**val** it **:** Quotations**.**Expr**<**Element**>** **=**

**<@** P **[**Text "This is a paragraph."**]**

**|>!** OnAfterRender **(fun** e **->** JavaScript**.**Alert**(**"Rendered!"**))** **@>**

**val** it **:** Quotations**.**Expr**<**Element**>** **=**

**[<**JavaScript**>]**

**let** CalculateResult a b **=** a**+**2**\***b

**val** CalculateResult **:** a**:**int **->** b**:**int **->** int

CalculateResult 10 20

**val** it **:** int **=** 50

**<@** CalculateResult 10 20 **@>**

**val** it **:** Quotations**.**Expr**<**int**>** **=**

CalculateResult 10 20

**val** it **:** int **=** 50

**<@** **let** foo **=** P **[**Text "foo"**]** **in** foo **@>**

**val** it **:** Quotations**.**Expr**<**Element**>** **=**

**[<**JavaScript**>]**

**let** Main **()** **=**

**let** element **=** HTML5**.**Tags**.**Canvas **[]**

**let** lastX**,** lastY**,** inLine **=** ref 0**,** ref 0**,** ref **false**

**let** labelPos **=** Span **[**Text "Position:"**]**

**let** updatePos **(**x**,** y**)** **=**

labelPos**.**Clear**()**

labelPos**.**Append **(**"X=" **+** string x **+** ", Y=" **+** string y**)**

**let** getXYFromMouseEvent **(**el**:** Element**)** **(**args**:** Events**.**MouseEvent**)** **=**

**let** pos **=** JQuery**.**JQuery**.**Of**(**el**.**Body**).**Position**()**

args**.**X **-** pos**.**Left**,** args**.**Y **-** pos**.**Top

**let** canvas **=** As**<**IntelliFactory**.**WebSharper**.**Html5**.**CanvasElement**>** element**.**Dom

canvas**.**Height **<-** 500

canvas**.**Width **<-** 500

**let** ctx **=** canvas**.**GetContext**(**"2d"**)**

Div **[**Attr**.**Style "text-align: center; cursor: crosshair"**]** **-<** **[**

labelPos

Br **[]**

element **-<** **[**Attr**.**Style "border: 1px solid gray"**]**

**|>!** OnMouseDown **(fun** el args **->**

**let** x**,** y **=** getXYFromMouseEvent el args

updatePos **(**x**,** y**)**

lastX **:=** x

lastY **:=** y

inLine **:=** **true**

ctx**.**BeginPath**()**

**)**

**|>!** OnMouseUp **(fun** el args **->**

inLine **:=** **false**

updatePos **<|** getXYFromMouseEvent el args

**)**

**|>!** OnMouseMove **(fun** el args **->**

**if** **!**inLine **then**

**let** x**,** y **=** getXYFromMouseEvent el args

updatePos **(**x**,** y**)**

ctx**.**MoveTo **(**float **!**lastX**,** float **!**lastY**)**

ctx**.**LineTo **(**float x**,** float y**)**

ctx**.**Stroke**()**

lastX **:=** x

lastY **:=** y

**else**

updatePos **<|** getXYFromMouseEvent el args

**)**

**|>!** OnMouseEnter **(fun** el args **->**

updatePos **<|** getXYFromMouseEvent el args

**)**

**|>!** OnMouseLeave **(fun** **\_** **\_** **->**

inLine **:=** **false**

**)**

P **[**B **[**Text "Draw with your mouse"**]]**

**]**

**<@** Main **()** **@>**

**val** Main **:** unit **->** Element

**val** it **:** Quotations**.**Expr**<**Element**>** **=**

**#**r "MyDir/MyAsm.dll"

**#**r "bin/net45/MyProject.dll"

NuGet**.**InstallPackage "WebSharper.JQueryMobile"

**#**r "IntelliFactory.WebSharper.JQuery.Mobile.dll"

Retrieving NuGet package**...**

Package retrieved**.**

Extracted assemblies available **for** referencing are**:**

**#**r "IntelliFactory.WebSharper.JQuery.Mobile.dll"

Done**.**

**val** it **:** unit **=** **()**

it

**val** it **:** unit **=** **()**

**//** Install **and** reference FSharp**.**Data**.**

NuGet**.**InstallPackage "FSharp.Data"

**#**r "FSharp.Data.dll"

**//** Instantiate the World Bank Type Provider**.**

**let** data **=** FSharp**.**Data**.**WorldBankData**.**GetDataContext**()**

**//** Load **and** chart data **of** interest**.**

**let** countries **=**

**[**

data**.**Countries**.**``United Kingdom``

data**.**Countries**.**``United States``

data**.**Countries**.**France

**]**

**let** ``School Enrollment`` **=**

Charting**.**Combine **[**

**for** country **in** countries **->**

country**.**Indicators**.**``School enrollment**,** tertiary **(%** gross**)**``

**|>** Seq**.**map **(fun** **(**year**,** value**)** **->** **(**string year**,** value**))**

**|>** Charting**.**Line

**|>** Charting**.**WithLegend country**.**Name

**]**

**|>** Charting**.**WithXTitle "School enrollment, tertiary (% gross)"

Retrieving NuGet package**...**

Package retrieved**.**

Extracted assemblies available **for** referencing are**:**

**#**r "FSharp.Data.DesignTime.dll"

**#**r "FSharp.Data.dll"

Done**.**

**val** it **:** unit **=** **()**

**//** Install **and** reference FSharp**.**Data**.**

NuGet**.**InstallPackage "FSharp.Data"

**#**r "FSharp.Data.dll"

**//** Instantiate the World Bank Type Provider**.**

**let** data **=** FSharp**.**Data**.**WorldBankData**.**GetDataContext**()**

**//** Load **and** chart data **of** interest**.**

**let** countries **=**

**[**

data**.**Countries**.**``United Kingdom``

data**.**Countries**.**``United States``

data**.**Countries**.**France

**]**

**let** ``School Enrollment`` **=**

Charting**.**Combine **[**

**for** country **in** countries **->**

country**.**Indicators**.**``School enrollment**,** tertiary **(%** gross**)**``

**|>** Seq**.**map **(fun** **(**year**,** value**)** **->** **(**string year**,** value**))**

**|>** Charting**.**Line

**|>** Charting**.**WithLegend country**.**Name

**]**

**|>** Charting**.**WithXTitle "School enrollment, tertiary (% gross)"

**val** data **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**WorldBankDataService

**val** countries **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**Country list **=**

**[**United Kingdom**;** United States**;** France**]**

**val** **(** School Enrollment **)** **:** Charting**.**Chart **=**

Retrieving NuGet package**...**

## F# Interactive part 2

Package retrieved**.**

Extracted assemblies available **for** referencing are**:**

**#**r "FSharp.Data.DesignTime.dll"

Couldn't copy FSharp**.**Data**.**DesignTime**.**dll**;** is it already installed **and** **in** use**?**

**#**r "FSharp.Data.dll"

Couldn't copy FSharp**.**Data**.**dll**;** is it already installed **and** **in** use**?**

Done**.**

**val** it **:** unit **=** **()**

**val** data **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**WorldBankDataService

**val** countries **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**Country list **=**

**[**United Kingdom**;** United States**;** France**]**

**val** **(** School Enrollment **)** **:** Charting**.**Chart **=**

**//** Install **and** reference FSharp**.**Data**.**

NuGet**.**InstallPackage "FSharp.Data"

**#**r "FSharp.Data.dll"

**//** Instantiate the World Bank Type Provider**.**

**let** data **=** FSharp**.**Data**.**WorldBankData**.**GetDataContext**()**

**//** Load **and** chart data **of** interest**.**

**let** countries **=**

**[**

data**.**Countries**.**``United Kingdom``

data**.**Countries**.**``United States``

data**.**Countries**.**France

**]**

**let** ``School Enrollment`` **=**

Charting**.**Combine **[**

**for** country **in** countries **->**

country**.**Indicators**.**``School enrollment**,** tertiary **(%** gross**)**``

**|>** Seq**.**map **(fun** **(**year**,** value**)** **->** **(**string year**,** value**))**

**|>** Charting**.**Line

**|>** Charting**.**WithLegend country**.**Name

**]**

**|>** Charting**.**WithXTitle "School enrollment, tertiary (% gross)"

Retrieving NuGet package**...**

Package retrieved**.**

Extracted assemblies available **for** referencing are**:**

**#**r "FSharp.Data.DesignTime.dll"

Couldn't copy FSharp**.**Data**.**DesignTime**.**dll**;** is it already installed **and** **in** use**?**

**#**r "FSharp.Data.dll"

Couldn't copy FSharp**.**Data**.**dll**;** is it already installed **and** **in** use**?**

Done**.**

**val** it **:** unit **=** **()**

**val** data **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**WorldBankDataService

**val** countries **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**Country list **=**

**[**United Kingdom**;** United States**;** France**]**

**val** **(** School Enrollment **)** **:** Charting**.**Chart **=**

**//** Install **and** reference FSharp**.**Data**.**

NuGet**.**InstallPackage "FSharp.Data"

**#**r "FSharp.Data.dll"

**//** Instantiate the World Bank Type Provider**.**

**let** data **=** FSharp**.**Data**.**WorldBankData**.**GetDataContext**()**

**//** Load **and** chart data **of** interest**.**

**let** countries **=**

**[**

data**.**Countries**.**``United Kingdom``

data**.**Countries**.**``United States``

data**.**Countries**.**France

**]**

**let** ``School Enrollment`` **=**

Charting**.**Combine **[**

**for** country **in** countries **->**

country**.**Indicators**.**``School enrollment**,** tertiary **(%** gross**)**``

**|>** Seq**.**map **(fun** **(**year**,** value**)** **->** **(**string year**,** value**))**

**|>** Charting**.**Line

**|>** Charting**.**WithLegend country**.**Name

**]**

**|>** Charting**.**WithXTitle "School enrollment, tertiary (% gross)"

Retrieving NuGet package**...**

Package retrieved**.**

Extracted assemblies available **for** referencing are**:**

**#**r "FSharp.Data.DesignTime.dll"

Couldn't copy FSharp**.**Data**.**DesignTime**.**dll**;** is it already installed **and** **in** use**?**

**#**r "FSharp.Data.dll"

Couldn't copy FSharp**.**Data**.**dll**;** is it already installed **and** **in** use**?**

Done**.**

**val** it **:** unit **=** **()**

**val** data **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**WorldBankDataService

**val** countries **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**Country list **=**

**[**United Kingdom**;** United States**;** France**]**

**val** **(** School Enrollment **)** **:** Charting**.**Chart **=**

**//** Install **and** reference FSharp**.**Data**.**

NuGet**.**InstallPackage "FSharp.Data"

**#**r "FSharp.Data.dll"

**//** Instantiate the World Bank Type Provider**.**

**let** data **=** FSharp**.**Data**.**WorldBankData**.**GetDataContext**()**

**//** Load **and** chart data **of** interest**.**

**let** countries **=**

**[**

data**.**Countries**.**``United Kingdom``

data**.**Countries**.**``United States``

data**.**Countries**.**France

**]**

**let** ``School Enrollment`` **=**

Charting**.**Combine **[**

**for** country **in** countries **->**

country**.**Indicators**.**``School enrollment**,** tertiary **(%** gross**)**``

**|>** Seq**.**map **(fun** **(**year**,** value**)** **->** **(**string year**,** value**))**

**|>** Charting**.**Line

**|>** Charting**.**WithLegend country**.**Name

**]**

**|>** Charting**.**WithXTitle "School enrollment, tertiary (% gross)"

Retrieving NuGet package**...**

Package retrieved**.**

Extracted assemblies available **for** referencing are**:**

**#**r "FSharp.Data.DesignTime.dll"

Couldn't copy FSharp**.**Data**.**DesignTime**.**dll**;** is it already installed **and** **in** use**?**

**#**r "FSharp.Data.dll"

Couldn't copy FSharp**.**Data**.**dll**;** is it already installed **and** **in** use**?**

Done**.**

**val** it **:** unit **=** **()**

**val** data **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**WorldBankDataService

**val** countries **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**Country list **=**

**[**United Kingdom**;** United States**;** France**]**

**val** **(** School Enrollment **)** **:** Charting**.**Chart **=**

**//** Install **and** reference FSharp**.**Data**.**

NuGet**.**InstallPackage "FSharp.Data"

**#**r "FSharp.Data.dll"

**//** Instantiate the World Bank Type Provider**.**

**let** data **=** FSharp**.**Data**.**WorldBankData**.**GetDataContext**()**

**//** Load **and** chart data **of** interest**.**

**let** countries **=**

**[**

data**.**Countries**.**``United Kingdom``

data**.**Countries**.**``United States``

data**.**Countries**.**France

**]**

**let** ``School Enrollment`` **=**

Charting**.**Combine **[**

**for** country **in** countries **->**

country**.**Indicators**.**``School enrollment**,** tertiary **(%** gross**)**``

**|>** Seq**.**map **(fun** **(**year**,** value**)** **->** **(**string year**,** value**))**

**|>** Charting**.**Line

**|>** Charting**.**WithLegend country**.**Name

**]**

**|>** Charting**.**WithXTitle "School enrollment, tertiary (% gross)"

Retrieving NuGet package**...**

Package retrieved**.**

Extracted assemblies available **for** referencing are**:**

**#**r "FSharp.Data.DesignTime.dll"

Couldn't copy FSharp**.**Data**.**DesignTime**.**dll**;** is it already installed **and** **in** use**?**

**#**r "FSharp.Data.dll"

Couldn't copy FSharp**.**Data**.**dll**;** is it already installed **and** **in** use**?**

Done**.**

**val** it **:** unit **=** **()**

**val** data **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**WorldBankDataService

**val** countries **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**Country list **=**

**[**United Kingdom**;** United States**;** France**]**

**val** **(** School Enrollment **)** **:** Charting**.**Chart **=**

**//** Install **and** reference FSharp**.**Data**.**

NuGet**.**InstallPackage "FSharp.Data"

**#**r "FSharp.Data.dll"

**//** Instantiate the World Bank Type Provider**.**

**let** data **=** FSharp**.**Data**.**WorldBankData**.**GetDataContext**()**

**//** Load **and** chart data **of** interest**.**

**let** countries **=**

**[**

data**.**Countries**.**``United Kingdom``

data**.**Countries**.**``United States``

data**.**Countries**.**France

**]**

**let** ``School Enrollment`` **=**

Charting**.**Combine **[**

**for** country **in** countries **->**

country**.**Indicators**.**``School enrollment**,** tertiary **(%** gross**)**``

**|>** Seq**.**map **(fun** **(**year**,** value**)** **->** **(**string year**,** value**))**

**|>** Charting**.**Line

**|>** Charting**.**WithLegend country**.**Name

**]**

**|>** Charting**.**WithXTitle "School enrollment, tertiary (% gross)"

Retrieving NuGet package**...**

Package retrieved**.**

Extracted assemblies available **for** referencing are**:**

**#**r "FSharp.Data.DesignTime.dll"

Couldn't copy FSharp**.**Data**.**DesignTime**.**dll**;** is it already installed **and** **in** use**?**

**#**r "FSharp.Data.dll"

Couldn't copy FSharp**.**Data**.**dll**;** is it already installed **and** **in** use**?**

Done**.**

**val** it **:** unit **=** **()**

**val** data **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**WorldBankDataService

**val** countries **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**Country list **=**

**[**United Kingdom**;** United States**;** France**]**

**val** **(** School Enrollment **)** **:** Charting**.**Chart **=**

**//** Install **and** reference FSharp**.**Data**.**

NuGet**.**InstallPackage "FSharp.Data"

**#**r "FSharp.Data.dll"

**//** Instantiate the World Bank Type Provider**.**

**let** data **=** FSharp**.**Data**.**WorldBankData**.**GetDataContext**()**

**//** Load **and** chart data **of** interest**.**

**let** countries **=**

**[**

data**.**Countries**.**``United Kingdom``

data**.**Countries**.**``United States``

data**.**Countries**.**Hungary

**]**

**let** ``School Enrollment`` **=**

Charting**.**Combine **[**

**for** country **in** countries **->**

country**.**Indicators**.**``School enrollment**,** tertiary **(%** gross**)**``

**|>** Seq**.**map **(fun** **(**year**,** value**)** **->** **(**string year**,** value**))**

**|>** Charting**.**Line

**|>** Charting**.**WithLegend country**.**Name

**]**

**|>** Charting**.**WithXTitle "School enrollment, tertiary (% gross)"

Retrieving NuGet package**...**

Package retrieved**.**

Extracted assemblies available **for** referencing are**:**

**#**r "FSharp.Data.DesignTime.dll"

Couldn't copy FSharp**.**Data**.**DesignTime**.**dll**;** is it already installed **and** **in** use**?**

**#**r "FSharp.Data.dll"

Couldn't copy FSharp**.**Data**.**dll**;** is it already installed **and** **in** use**?**

Done**.**

**val** it **:** unit **=** **()**

**val** data **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**WorldBankDataService

**val** countries **:** FSharp**.**Data**.**WorldBankData**.**ServiceTypes**.**Country list **=**

**[**United Kingdom**;** United States**;** Hungary**]**

**val** **(** School Enrollment **)** **:** Charting**.**Chart **=**

## Módosított sitelet

namespace Application

**open** IntelliFactory**.**Html

**open** IntelliFactory**.**WebSharper

**open** IntelliFactory**.**WebSharper**.**Sitelets

**type** Action **=**

**|** **[<**CompiledName "haza"**>]** Home

**|** **[<**CompiledName "öbáut"**>]** About

**|** **[<**CompiledName "ekhó"**>]** Echo **of** string

**module** Controls **=**

**[<**Sealed**>]**

**type** EntryPoint**()** **=**

**inherit** Web**.**Control**()**

**[<**JavaScript**>]**

override \_\_**.**Body **=**

Client**.**Main**()** **:>** **\_**

**module** Skin **=**

**open** System**.**Web

**type** Page **=**

**{**

Title **:** string

Body **:** list**<**Content**.**HtmlElement**>**

**}**

**let** MainTemplate **=**

Content**.**Template**<**Page**>(**"~/Main.html"**)**

**.**With**(**"title"**,** **fun** x **->** x**.**Title**)**

**.**With**(**"body"**,** **fun** x **->** x**.**Body**)**

**let** WithTemplate title body **:** Content**<**Action**>** **=**

Content**.**WithTemplate MainTemplate **<|** **fun** context **->**

**{**

Title **=** title

Body **=** body context

**}**

**module** Site **=**

**let** **(** **=>** **)** text url **=**

A **[**HRef url**]** **-<** **[**Text text**]**

**let** Links **(**ctx**:** Context**<**Action**>)** **=**

UL **[**

LI **[**"Home" **=>** ctx**.**Link Home**]**

LI **[**"About" **=>** ctx**.**Link About**]**

LI **[**"Echo" **=>** ctx**.**Link **(**Action**.**Echo "teszt"**)]**

**]**

**let** HomePage **=**

Skin**.**WithTemplate "HomePage" **<|** **fun** ctx **->**

**[**

Div **[**Text "HOME"**]**

Div **[new** Controls**.**EntryPoint**()]**

Links ctx

**]**

**let** AboutPage **=**

Skin**.**WithTemplate "AboutPage" **<|** **fun** ctx **->**

**[**

Div **[**Text "ABOUT"**]**

Links ctx

**]**

**let** EchoPage s **=**

Skin**.**WithTemplate "EchoPage" **<|** **fun** ctx **->**

**[**

Div **[**Text **<|** "ECHO: " **+** s**]**

Links ctx

**]**

**let** Main **=**

**let** site1**=**

Sitelet**.**Infer **(** **function**

**|** Action**.**About **->** AboutPage

**|** Action**.**Home **->** HomePage

**|** Action**.**Echo s **->** EchoPage s

**)**

Sitelet**.**Sum **[**

Sitelet**.**Content "/" Action**.**Home HomePage

site1

**]**

**[<**Sealed**>]**

**type** Website**()** **=**

interface IWebsite**<**Action**>** **with**

member this**.**Sitelet **=** Site**.**Main

member this**.**Actions **=** **[**Home**;** About**]**

**type** Global**()** **=**

**inherit** System**.**Web**.**HttpApplication**()**

member g**.**Application\_Start**(**sender**:** obj**,** args**:** System**.**EventArgs**)** **=**

**()**

**[<**assembly**:** Website**(**typeof**<**Website**>)>]**

**do** **()**

Előadás vége.