

Type providers for Kotlin

Student: Anya Yaveyn

Advisor: Marat Ahin

Goal

```
tprs.fs x
1  module tprs
2
3  open FSharp.Data
4
5  type Simple = JsonProvider<"" { "name":"John", "age":94 } "">
6
7  [
```

Goal

```
// Simple type wrapping CSV data
type CsvFile(filename) =
// Cache the sequence of all data lines (all lines but the first)
let data =
seq { for line in File.ReadAllLines(filename) |> Seq.skip 1 do
yield line.Split(',') |> Array.map float }
|> Seq.cache
member __.Data = data

[<TypeProvider>]
type public MiniCsvProvider(cfg:TypeProviderConfig) as this =
inherit TypeProviderForNamespaces()

// Get the assembly and namespace used to house the provided types.
let asm = System.Reflection.Assembly.GetExecutingAssembly()
let ns = "Samples.FSharp.MiniCsvProvider"

// Create the main provided type.
let csvTy = ProvidedTypeDefinition(asm, ns, "MiniCsv", Some(typeof<obj>))

// Parameterize the type by the file to use as a template.
let filename = ProvidedStaticParameter("filename", typeof<string>)
do csvTy.DefineStaticParameters([filename], fun tyName [| :? string as filename |] ->

// Resolve the filename relative to the resolution folder.
let resolvedFilename = Path.Combine(cfg.ResolutionFolder, filename)
```

Goal

- Type providers like in F#

Goal

- **JSON** provider like in F#
- Type providers like in F#

Result

- **JSON** provider like in F# *

Result

```
1
2 package test
3
4 provided class User {
5     companion object {
6         val model = ""{"name":"Bilbo Baggins", "age":111}""
7     }
8 }
9
```

```
2
3 import test.User
4
5 fun main(args: Array<String>) {
6
7     val person = User()
8
9     person.name = "Taras Bulba"
10    person.age = 52
11
12    println(person.name!!) //=> Taras Bulba
13    println(person.age!!) //=> 52
14
15 }
16
```

Result

```
1
2  package test
3
4  provided class User {
5      companion object {
6          val model = """"{"name":"Bilbo Baggins", "age":111}""""
7      }
8  }
```

```
2
3  import test.User
4
5  fun main(args: Array<String>) {
6
7      val person = User("""{"name":"Taras Bulba", "age":52}""")
8
9      println(person.name)           //=> null
10     println(person.age)            //=> null
11
12 }
13
```


Challenges

- Kotlin
- Kotlin source code
- ~~Java annotations~~
- Ant, Gradle and Maven in one project
- F# type providers

Thank you for listening

https://github.com/bronti/kotlin/tree/type_providers