

014: 単純なケースで `deriving Show` をサポートする

↑ up

- issued: 2020-04-12
- 分類: B 機能追加
- status: Closed (2020-04-12)

説明

`deriving Show` のサポート ⇒ `deriving` ではなく、まず明示的に `Show` クラスにいれるやり方を先に対処することにした。

調査ログ

2020-04-12 (Sun)

現状確認

この件のために、いくつかテストケースを追加した。
 `drvshow.hs`: これは、 `print` が無いとってこける。

```
data Hoge = Hoge Integer
          | Fuga [Char]
          deriving Show
```

```
x = Hoge 10
y = Fuga "nine"
```

```
main = do
  putStrLn $ show x
  print y
```

`drvshow0.hs`: これは、コンパイルは通ってしまっ、ランタイムで警告になる。

```
data Hoge = Hoge Integer
          | Fuga [Char]
          deriving Show
```

```
x = Hoge 10
```

```
y = Fuga "nine"
```

```
main = do
  putStrLn $ show x
  putStrLn $ show y
```

drvshowlerr.hs : というわけで、現状では、これがコンパイルエラーにならない (なるべき)

```
data Hoge = Hoge Integer
          | Fuga [Char]
```

```
x = Hoge 10
y = Fuga "nine"
```

```
main = do
  putStrLn $ show x
  putStrLn $ show y
```

実装

すでに、デフォルトの show ファンクションにあたるものの原型は、ランタイムにあったので、それを手直しして実装。また、lib/Prelude.hs に print 関数も追加した。以下の二つのケースが通るようになった :

- sample138
- sample139

実装していて気になったのが、Prim.java にある ShowFunc.call の引数が、なぜ AtomExpr ではなく、AtomExpr [] (配列) になっているのかということ。あとで調べておこう。⇒ 016

また、現状では deriving Show してなくても、show できてしまう。これは、型検査の問題なのだが、本件とはわけて別の issue とする。⇒ 017