

IO Monad - ZIO



Example ZIO program

```
object LastItem {  
  def main(args: Array[String]): Unit = {  
    val runtime = new LiveRuntime {}  
  
    val showLastItem = for (  
      maxItemResponse <- httpClient.get(getMaxItemURL);  
      maxItem <- parseMaxItemResponse(maxItemResponse);  
      itemResponse <- httpClient.get(getItemURL(maxItem));  
      item <- parseItemResponse(itemResponse);  
      _ <- showComment(item)  
    ) yield ()  
  
    val program = showLastItem.repeat(Schedule.spaced(10.seconds))  
  
    runtime.unsafeRunSync(program)  
  }  
}
```

IO Monad - ZIO



Sample environment module

```
trait HttpClient {  
  val httpClient : HttpClient.Service[Any with HttpClient with Blocking]  
}  
object HttpClient {  
  trait Service[R <: HttpClient with Blocking] {  
    def get(url: String): ZIO[R, Throwable, String]  
  }  
}
```