

- History
 - Development began in September of 2003
 - Began as embedded scripting language hosted on the JVM
 - Focus shifted as features were added incrementally
 - Moved away from JVM to native code generation
 - Today has extensive user libraries
- Basic syntax & semantics
 - The stack & postfix notation
 - Postfix example (RPN with a stack, 1 2 + sort of thing)
 - Other stack-based languages
 - Parsing algorithm
 - * "
 - * token
 - ordinary word (terminology, incl. “vocabulary”)
 - parsing word
 - * number
 - Example ordinary words: stack shufflers
 - Example parsing words: quotations
- Concatenative programming
 - “Pipeline code”
 - Whitespace = function composition
 - Point-free style
 - Origin of name “Factor”
- Stack effects
 - Basic stack effects: stack shufflers illustrated
 - Complex stack effects: row polymorphism & types
 - Stack checker
- Control flow
 - Combinators
 - * if
 - * each
 - * while
 - Dataflow combinators
 - * Dip/keep

- * Cleave
 - * Spread
 - * Apply
- Object system
 - tuples
 - generics & methods
- Libraries & metaprogramming
 - Results of evolution
 - locals?
 - fry?
 - macros?
 - functors?
 - ffi?