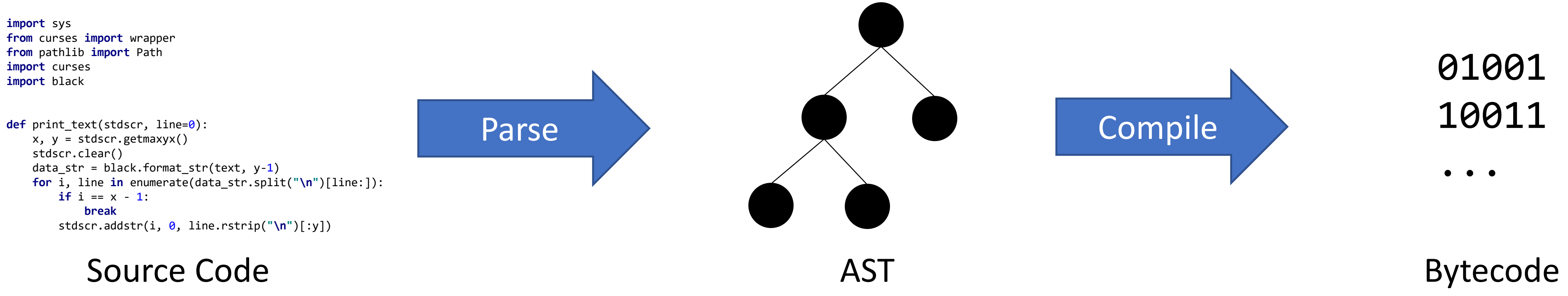
Python AST by example

Felix Kohlgrüber

Abstract Syntax Tree



```
import sys
from curses import wrapper
from pathlib import Path
import curses
import black


def print_text(stdscr, line=0):
    x, y = stdscr.getmaxyx()
    stdscr.clear()
    data_str = black.format_str(text, y-1)
    for i, line in enumerate(data_str.split("\n")[line:]):
        if i == x - 1:
            break
        stdscr.addstr(i, 0, line.rstrip("\n")[:y])
```

Abstract Grammar

```
mod = Module(stmt* body)
    ...

stmt = FunctionDef(identifier name,
                   arguments args,
                   stmt* body, ...)
      | Assign(expr* targets, expr value)
      | If(expr test, stmt* body, stmt* orelse)
      | Expr(expr value)
      ...
```

```
expr = BinOp(expr left, operator op, expr right)
     | Call(expr func, expr* args, keyword* keywords)
     | Num(object n)
     | Str(string s)
     | List(expr* elts, expr_context ctx)
     ...

boolop = And | Or

operator = Add | Sub | Mult | Div | ...

...
```

https://docs.python.org/3/library/ast.html#abstract-grammar

Demo