file_input () ENDMARKER (continued_file_input () continued_file_input () stmt () continued_file_input () stmt () simple_stmt () NEWLINE (stmt () continued_file_input () compound_stmt () small_stmt () $next_small_stmt(\epsilon)$ for_stmt () simple_stmt () simple_stmt () FOR (for) testlist () small_stmt () COLON (:) suite () testlist_star_expr () $next_small_stmt$ (ϵ) else_stmt (ϵ) continued_expr_stmt (ϵ) INDENT (-) stmts () DEDENT (-) continued_testlist (ϵ) $next_small_stmt$ (ϵ) test () continued_comma_test (ε) (test () expr_stmt () expr () continued_exprlist (ε) small_stmt () stmt () expr_stmt() testlist_star_expr () or_test() or_test () xor_expr () continued_xor_expr (ϵ) continued_expr_stmt (ϵ) testlist_star_expr () and_expr () simple_stmt () continued_expr_stmt () continued_testlist (ϵ) and_test () and_test () continued_and_expr (ϵ) continued_or_test (ε) continued_or_test (ε) test () NEWLINE (EQUAL (=) shift_expr () or_test () continued_shift_expr (ϵ) small_stmt () continued_and_test (ϵ) continued_testlist (ϵ) testlist_star_expr () not_test () or_test () continued_testlist (ϵ) and_test () continued_arith_expr (ϵ) expr_stmt () continued_or_test (ϵ) arith_expr () comparison () comparison () or_test() not_test () continued_and_test (ϵ) term () continued_term (ε) expr () testlist_star_expr () comparison_stmt (ϵ) comparison_stmt (ϵ) continued_expr_stmt (ϵ) continued_testlist (ε) factor () and_test () continued_factor (ε) (test () not_test () continued_or_test (ε) comparison () continued_xor_expr (ϵ) xor_expr () xor_expr () continued_xor_expr (ϵ) continued_and_test (ϵ) power () or_test () continued_and_expr (ϵ) comparison () not_test () expr () comparison_stmt (ϵ) and_expr () continued_and_expr (ϵ) and_expr() comparison_stmt (ε) atom_expr() shift_expr() continued_shift_expr (ϵ) and_test () continued_or_test (ϵ) shift_expr () continued_shift_expr (ε) (expr () comparison () xor_expr () continued_xor_expr (ϵ) and_expr () atom () arith_expr () continued_and_test (ϵ) arith_expr () continued_arith_expr (ϵ) expr() comparison_stmt (ϵ) continued_and_expr (ϵ) trailers (ε) not_test () xor_expr () continued_xor_expr (ϵ) continued_arith_expr (ϵ) NAME (byte) (term () shift_expr () continued_shift_expr (ϵ) continued_term (ϵ) comparison () continued_term (ε) and_expr () continued_and_expr (ϵ) (term () xor_expr () continued_xor_expr (ϵ) continued_arith_expr (ε) factor () continued_shift_expr (ϵ) continued_and_expr (ϵ) continued_factor (ε) comparison_stmt (ϵ) factor () continued_factor (ε) shift_expr () and_expr () arith_expr () expr () arith_expr () continued_arith_expr (ϵ) shift_expr () power () continued_shift_expr (ϵ) term () continued_term (ε) xor_expr () continued_xor_expr (ϵ) power () term () arith_expr () continued_arith_expr (ϵ) factor () continued_factor (ε) and_expr() continued_and_expr (ϵ) atom_expr () continued_term (ϵ) atom_expr () factor () continued_factor (ε) term () power () trailers (ε) shift_expr () continued_shift_expr (ϵ) trailers () continued_term (ε) atom () atom () NAME (ascii_bytes) NAME (print) factor () continued_factor (ϵ) (arith_expr () continued_arith_expr (ϵ) $\left(\text{trailers (e)} \right) \left(\text{trailer (())} \right)$ atom_expr() power () power () trailers () continued_term (ε) atom () atom_expr() (term () continued_factor (ε) trailers (ε) NAME (print) trailers (ε) trailer () factor () atom () atom_expr() CLOSING_BRACKET ()) OPENING_BRACKET (() NAME (ascii_bytes) arglist () atom () trailers (ε) power () strings () continued_argument (ϵ) atom_expr() argument () STRING (b'Hello, World!') test () trailers () atom () NAME (print) trailer () or_test() trailers (ε) OPENING_BRACKET (() arglist () CLOSING_BRACKET ()) continued_or_test (ε) continued_and_test (ε) not_test () continued_argument () argument () COMMA (,) (test () comparison () continued_argument (ϵ) argument () or_test () EQUAL (=) (test () (test () (expr () comparison_stmt (ϵ) or_test () or_test () and_test () continued_or_test (ε) continued_xor_expr (ε) xor_expr () not_test() continued_and_expr (ϵ) continued_and_test (ε) and_test () and_test () continued_or_test (ε) continued_or_test (ε) continued_shift_expr (ε) shift_expr () not_test () continued_and_test (ϵ) not_test () continued_and_test (ϵ) comparison () continued_arith_expr (ϵ) comparison_stmt (ε) arith_expr () expr () comparison () comparison () comparison_stmt (ε) comparison_stmt (ε) continued_term (ε) expr () expr () xor_expr () continued_xor_expr (ϵ) factor () continued_factor (ε) and_expr () continued_and_expr (ϵ) xor_expr () continued_xor_expr (ϵ) xor_expr () continued_xor_expr (ϵ) continued_shift_expr (ϵ) shift_expr () and_expr() power () continued_and_expr (ε) and_expr () continued_and_expr (ϵ) arith_expr () continued_arith_expr (ϵ) shift_expr () continued_shift_expr (ε) shift_expr () continued_shift_expr (ϵ) atom_expr() continued_arith_expr (ϵ) trailers (ε) term () arith_expr () continued_arith_expr (ϵ) arith_expr () continued_term (ϵ) NAME (ascii_bytes) factor () continued_factor (ε) term () (term () continued_term (ε) continued_term (ε) factor () continued_factor (ε) continued_factor (ε) power () factor () atom_expr () power () power () atom () trailers (ε) atom_expr () atom_expr () NAME (byte) atom () trailers (ε) atom () trailers (ε) NAME (end)

start ()