Solutions

2.1

- Every later diagnostic (parse and semantic errors) cites them ✓.

2.2

- `ident` – the longest lexeme wins; `ifx` is one IDENT ✓.

2.3

- Whitespace and comments ✓.

3.1

- IDENT(total)@1, ASSIGN(=)@7, IDENT(ifx)@9, PLUS(+)@13, NUM(42)@15, SEMI(;)@17 ✓.
- `ifx`: the ident regex matches 3 characters, `if` matches 2 – longest wins; priority would matter only on exact ties ✓.
- Under our spec `42abc` is NUM(42) then IDENT(abc) – reported as a lexical error (adjacent value tokens with no separator) ✓.
- Without keywords, nothing changes on THIS line – `ifx` stays IDENT; `if` alone would change from KW to IDENT ✓.

3.2

- Report-as-error: the lexer refuses and no parser state is built ✓.
- Emit-both: NUM then IDENT reaches the parser, which will choke on two adjacent value tokens ✓.
- Identifiers cannot start with a digit here, so `12abc` cannot be one token of either class – the error policy matches the language ✓.
- Principle: the error policy is part of the lexer's specification – chosen, not discovered ✓.