Quiz

- Question #1: What is the value of the following expression?

```
{+ 1 2}
```

- Wrong answer: **0**
- Wrong answer: **42**
- Answer: **3**

Quiz

- Question #2: What is the value of the following expression?

```
{+ fun 17 8}
```

- Wrong answer: **error**

- Answer: Trick question! `{+ fun 17 8}` is not an expression

Language Grammar for Quiz

```
<MFAE>  ::=  <num>
          |  true
          |  false
          |  {+ <MFAE> <MFAE>}
          |  {- <MFAE> <MFAE>}
          |  {= <MFAE> <MFAE>}
          |  <id>
          |  {fun {<id>*} <MFAE>}
          |  {<MFAE> <MFAE>*}
          |  {if <MFAE> <MFAE> <MFAE>}
```

Quiz

- Question #3: Is the following an expression?

```
{{fun {} 1} 7}
```

- Wrong answer: **No**

- Answer: **Yes** (according to our grammar)

Quiz

- Question #4: What is the value of the following expression?

```
{{fun {} 1} 7}
```

- Answer: **1** (according to some interpreters)

- But no *real* language would accept `{{fun {} 1} 7}`

- Let's agree to call `{{fun {} 1} 7}` an ***ill-formed expression*** because `{fun {} 1}` should be used with only zero arguments

- Let's agree to never evaluate ill-formed expressions

Quiz

- Question #5: What is the value of the following expression?

```
{{fun {} 1} 7}
```

- Answer: **None** - the expression is ill-formed

Quiz

- Question #6: Is the following a well-formed expression?

```
{+ {fun {} 1} 8}
```

- Answer: **Yes**

Quiz

- Question #7: What is the value of the following expression?

```
{+ {fun {} 1} 8}
```

- Answer: **None** - it produces an error:

+: expects a numV, given a closureV

- Let's agree that a `fun` expression cannot be inside a `+` form

Quiz

- Question #8: Is the following a well-formed expression?

```
{+ {fun {} 1} 8}
```

- Answer: **No**

Quiz

- Question #9: Is the following a well-formed expression?

```
{+ {{fun {x} x} 7} 5}
```

- Answer: Depends on what we meant by *inside* in our most recent agreement
  - *Anywhere inside* - **No**
  - *Immediately inside* - **Yes**
- Since our intrepreter produces **12**, and since that result makes sense, let's agree on *immediately inside*

Quiz

- Question #10: Is the following a well-formed expression?

```
{+ {{fun {x} x} {fun {y} y}} 5}
```

- Answer: **Yes**, but we don't want it to be!

Quiz

- Question #11: Is it possible to define ***well-formed*** (as a decidable property) so that we reject all expressions that produce errors?

- Answer: **Yes**: reject *all* expressions!

Quiz

- Question #12: Is it possible to define ***well-formed*** (as a decidable property) so that we reject *only* expressions that produce errors?

- Answer: **No**

```
{+ 1 {if ... 1 {fun {x} x}}}
```

- If we always knew whether `...` produces true or false, we could solve the halting problem

Types

- Solution to our dilemma
  - In the process of rejecting expressions that are certainly bad, also reject some expressions that are good

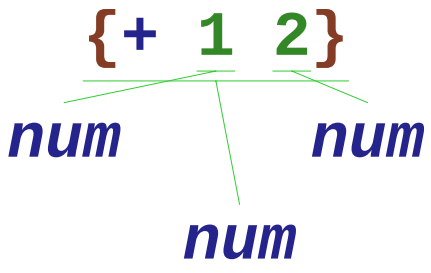
```
{+ 1 {if {prime? 131101} 1 {fun {x} x}}}
```

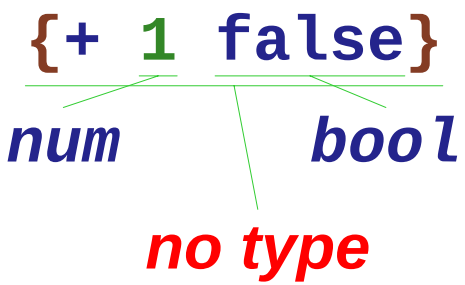
- Overall strategy:
  - Assign a ***type*** to each expression *without evaluating*
  - Compute the type of a complex expression based on the types of its subexpressions

Types

`1` : *num*

`true` : *bool*





Type Rules

<num> : *num*

true : *bool*

false : *bool*

$$\frac{\texttt{<MFAE>}_1 : \textit{num} \qquad \texttt{<MFAE>}_2 : \textit{num}}{\texttt{\{+ <MFAE>}_1\ \texttt{<MFAE>}_2\texttt{\}} : \textit{num}}$$

1 : *num*

true : *bool*

$$\frac{\texttt{1} : \textit{num} \qquad \texttt{2} : \textit{num}}{\texttt{\{+ 1 2\}} : \textit{num}}$$

$$\frac{\texttt{1} : \textit{num} \qquad \texttt{false} : \textit{bool}}{\texttt{\{+ 1 false\}} : \textit{no type}}$$

Type Rules

<num> : *num*

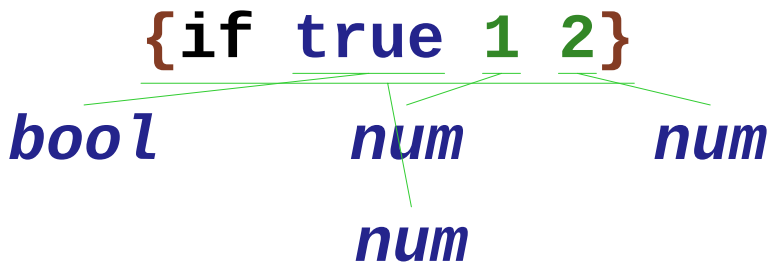
true : *bool*

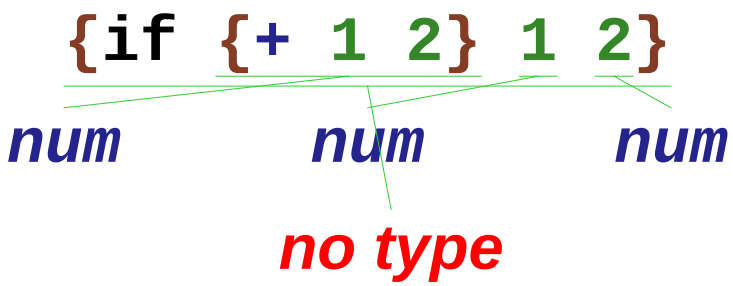
false : *bool*

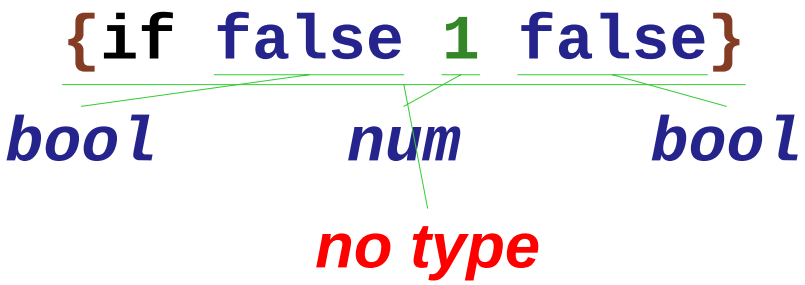
$$\frac{\texttt{<MFAE>}_1 : \textit{num} \qquad \texttt{<MFAE>}_2 : \textit{num}}{\texttt{\{+ <MFAE>}_1\ \texttt{<MFAE>}_2\texttt{\}} : \textit{num}}$$

$$\frac{\dfrac{\texttt{1} : \textit{num} \qquad \texttt{2} : \textit{num}}{\texttt{\{+ 1 2\}} : \textit{num}} \qquad \texttt{3} : \textit{num}}{\texttt{\{+ \{+ 1 2\} 3\}} : \textit{num}}$$

Types: Conditionals







Conditional Type Rules

$$\frac{\texttt{<MFAE>}_1 \texttt{ : } \textit{bool} \qquad \texttt{<MFAE>}_2 \texttt{ : <type>}_0 \qquad \texttt{<MFAE>}_3 \texttt{ : <type>}_0}{\texttt{\{if <MFAE>}_1 \texttt{ <MFAE>}_2 \texttt{ <MFAE>}_3\texttt{\} : <type>}_0}$$

$$\frac{\texttt{true : } \textit{bool} \qquad \texttt{1 : } \textit{num} \qquad \texttt{2 : } \textit{num}}{\texttt{\{if true 1 2\} : } \textit{num}}$$

$$\frac{\texttt{\{+ 1 2\} : } \textit{num} \qquad \texttt{1 : } \textit{num} \qquad \texttt{2 : } \textit{num}}{\texttt{\{if \{+ 1 2\} 1 2\} : } \textit{no type}}$$

$$\frac{\texttt{false : } \textit{bool} \qquad \texttt{1 : } \textit{num} \qquad \texttt{false : } \textit{bool}}{\texttt{\{if false 1 false\} : } \textit{no type}}$$

Types: Variables and Functions

`x` : *no type*

`{fun {x : bool} x}`

bool

(*bool* → *bool*)

`{fun {x : bool} {if x 1 2}}`

bool *num* *num*

num

(*bool* → *num*)

Variable and Function Type Rules

$$[\ \ldots\ \texttt{<id>} \leftarrow \tau\ \ldots\ ] \vdash \texttt{<id>} : \tau$$

$$\frac{\Gamma[\ \texttt{<id>} \leftarrow \tau_1\ ] \vdash \texttt{e} : \tau_2}{\Gamma \vdash \texttt{\{fun \{<id> : } \tau_1 \texttt{\} e\}} : (\tau_1 \rightarrow \tau_2)}$$

Abbreviations: τ = **<type>** **e** = **<MFAE>** Γ = **<env>**

Variable and Function Type Rules

$$[\ \ldots\ \texttt{<id>} \leftarrow \tau\ \ldots\ ] \vdash \texttt{<id>} : \tau$$

$$\frac{\Gamma[\ \texttt{<id>} \leftarrow \tau_1\ ] \vdash \texttt{e} : \tau_2}{\Gamma \vdash \{\texttt{fun}\ \{\texttt{<id>} : \tau_1\}\ \texttt{e}\} : (\tau_1 \rightarrow \tau_2)}$$

$$\emptyset \vdash \texttt{x} : \textit{no type}$$

$$\frac{[\ \texttt{x} \leftarrow \textit{bool}\ ] \vdash \texttt{x} : \textit{bool}}{\emptyset \vdash \{\texttt{fun}\ \{\texttt{x} : \texttt{bool}\}\ \texttt{x}\} : (\textit{bool} \rightarrow \textit{bool})}$$

$$\frac{\dfrac{[\ \texttt{x} \leftarrow \textit{bool}\ ] \vdash \texttt{x} : \textit{bool} \quad [\ \texttt{x} \leftarrow \textit{bool}\ ] \vdash \texttt{1} : \textit{num} \quad [\ \texttt{x} \leftarrow \textit{bool}\ ] \vdash \texttt{2} : \textit{num}}{[\ \texttt{x} \leftarrow \textit{bool}\ ] \vdash \{\texttt{if}\ \texttt{x}\ \texttt{1}\ \texttt{2}\} : \textit{num}}}{\emptyset \vdash \{\texttt{fun}\ \{\texttt{x} : \texttt{bool}\}\ \{\texttt{if}\ \texttt{x}\ \texttt{1}\ \texttt{2}\}\} : (\textit{bool} \rightarrow \textit{num})}$$

Revised Rules

$$\Gamma \vdash \texttt{<num>} : \textit{num}$$

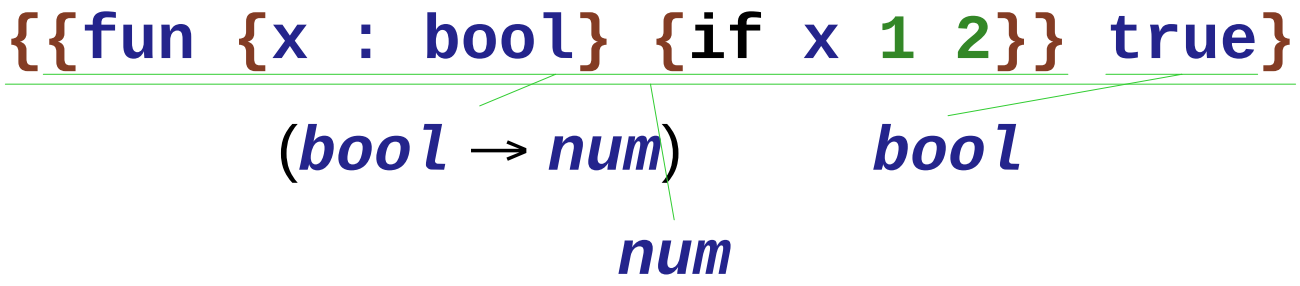
$$\Gamma \vdash \texttt{true} : \textit{bool}$$

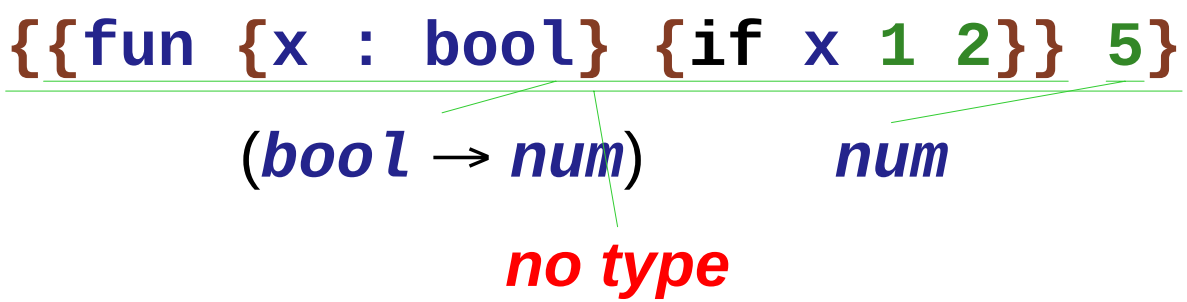
$$\Gamma \vdash \texttt{false} : \textit{bool}$$

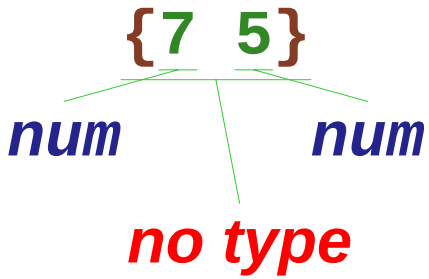
$$\frac{\Gamma \vdash \mathbf{e}_1 : \textit{num} \qquad \Gamma \vdash \mathbf{e}_2 : \textit{num}}{\Gamma \vdash \{\texttt{+}\ \mathbf{e}_1\ \mathbf{e}_2\} : \textit{num}}$$

$$\frac{\Gamma \vdash \mathbf{e}_1 : \textit{bool} \qquad \Gamma \vdash \mathbf{e}_2 : \tau_0 \qquad \Gamma \vdash \mathbf{e}_3 : \tau_0}{\Gamma \vdash \{\texttt{if}\ \mathbf{e}_1\ \mathbf{e}_2\ \mathbf{e}_3\} : \tau_0}$$

Types: Function Calls







Function Call Type Rule

$$\frac{\Gamma \vdash \mathbf{e}_1 \ : \ (\tau_2 \rightarrow \tau_3) \qquad \Gamma \vdash \mathbf{e}_2 \ : \ \tau_2}{\Gamma \vdash \{\mathbf{e}_1 \ \mathbf{e}_2\} \ : \ \tau_3}$$

$$\frac{\varnothing \vdash \texttt{\{fun \{x : bool\} \{if x 1 2\}\}} : (\textit{bool} \rightarrow \textit{num}) \qquad \varnothing \vdash \texttt{true} : \textit{bool}}{\varnothing \vdash \texttt{\{\{fun \{x : bool\} \{if x 1 2\}\} true\}} : \textit{num}}$$

$$\frac{\varnothing \vdash \texttt{\{fun \{x : bool\} \{if x 1 2\}\}} : (\textit{bool} \rightarrow \textit{num}) \qquad \varnothing \vdash \texttt{5} : \textit{num}}{\varnothing \vdash \texttt{\{\{fun \{x : bool\} \{if x 1 2\}\} 5\}} : \textit{no type}}$$

$$\frac{\varnothing \vdash \texttt{7} : \textit{num} \qquad \varnothing \vdash \texttt{5} : \textit{num}}{\varnothing \vdash \texttt{\{7 5\}} : \textit{no type}}$$

Types: Multiple Arguments

```
{fun {x : num y : num} {+ x y}}
```

num *num*

num

(*num num* → *num*)

```
{{fun {x : num y : num} {+ x y}} 5 6}
```

(*num num* → *num*) *num* *num*

num

```
{{fun {x : num y : num} {+ x y}} 5}
```

(*num num* → *num*) *num*

no type

Revised Function and Call Rules

$$\frac{\Gamma[\ \texttt{<id>}_1 \leftarrow \tau_1 \ \ldots\ \texttt{<id>}_n \leftarrow \tau_n\ ] \vdash \texttt{e} : \tau_0}{\Gamma \vdash \{\texttt{fun}\ \{\texttt{<id>}_1 : \tau_1\ \ldots\ \texttt{<id>}_n : \tau_n\}\ \texttt{e}\} : (\tau_1 \ldots \tau_n \rightarrow \tau_0)}$$

$$\frac{\Gamma \vdash \texttt{e}_0 : (\tau_1 \ldots \tau_n \rightarrow \tau_0) \qquad \Gamma \vdash \texttt{e}_1 : \tau_1 \qquad \ldots \qquad \Gamma \vdash \texttt{e}_n : \tau_n}{\Gamma \vdash \{\texttt{e}_0\ \texttt{e}_1\ \ldots\ \texttt{e}_n\} : \tau_0}$$