Term variables, x, y, z variables
Type variables, a, b type variables
Integers, i

Monotypes, τ	::=			Monotypes
	|	`Int`		
	|	a		
	|	$\tau_1 \to \tau_2$		

Rho-types, ρ	::=			Rho-types
	|	τ		
	|	(ρ)	S	

Polytypes, σ	::=			
	|	ρ		
	|	$\forall a.\sigma$	bind a in σ	
	|	$[a \mapsto \tau]\sigma$	M	

Terms, t, u	::=			
	|	i		Literal
	|	x		Variable
	|	$\lambda x.t$	bind x in t	Abstraction
	|	$t\ u$		Application
	|	$\lambda(x :: \sigma).t$	bind x in t	Typed abstraction
	|	`let` $x = u$ `in` t	bind x in t	Local binding
	|	$t :: \sigma$		Type annotation
	|	(t)	S	
	|	$t\,[x \rightsquigarrow t']$	M	

value, v	::=			
	|	i		
	|	$\lambda x.t$	bind x in t	
	|	$\lambda(x :: \sigma).t$	bind x in t	

Typing contexts, Γ	::=			
	|	ϵ		empty context
	|	$\Gamma, x : \sigma$		assumption

| *fv* | ::= | | | |
| | \| | $fv(t)$ | M | |

ftvany	::=			
	|	$\text{ftv}(\tau)$	M	
	|	$\text{ftv}(\rho)$	M	
	|	$\text{ftv}(\sigma)$	M	
	|	$\text{ftv}(\Gamma)$	M	

terminals	::=			
	|	$\longrightarrow$		
	|	$\to$		
	|	$::$		
	|	fv		
	|	ftv		

| | ∈
| | ∉
| | ⊢

formula   ::=
| *judgement*
| value v
| uniq Γ
| closed σ
| $a \notin ftvany$
| $x : \sigma \in \Gamma$

JTyping   ::=
| $\Gamma \vdash t : \sigma$

JStep   ::=
| $t \longrightarrow u$

judgement   ::=
| *JTyping*
| *JStep*

user_syntax   ::=
| Term variables
| Type variables
| Integers
| Monotypes
| Rho-types
| Polytypes
| Terms
| *value*
| Typing contexts
| *fv*
| *ftvany*
| *terminals*
| *formula*

$\boxed{\Gamma \vdash t : \sigma}$

$$\frac{}{\Gamma \vdash i : \texttt{Int}} \quad \text{TYP\_INT}$$

$$\frac{\begin{array}{c} \text{uniq}\,\Gamma \\ x : \sigma \in \Gamma \end{array}}{\Gamma \vdash i : \sigma} \quad \text{TYP\_VAR}$$

$$\frac{\Gamma, x : \tau_1 \vdash t : \tau_2}{\Gamma \vdash (\lambda x.t) : (\tau_1 \to \tau_2)} \quad \text{TYP\_ABS}$$

$$\frac{\begin{array}{c} \Gamma \vdash t : \tau_1 \to \tau_2 \\ \Gamma \vdash u : \tau_1 \end{array}}{\Gamma \vdash t\ u : \tau_2} \quad \text{TYP\_APP}$$

$$\frac{\begin{array}{c}\Gamma \vdash u : \sigma \\ \Gamma, x : \sigma \vdash t : \rho\end{array}}{\Gamma \vdash \mathtt{let}\ x = u\ \mathtt{in}\ t : \rho}\ \text{TYP\_LET}$$

$$\frac{\begin{array}{c}\mathsf{closed}\ \sigma \\ \Gamma \vdash t : \sigma\end{array}}{\Gamma \vdash (t :: \sigma) : \sigma}\ \text{TYP\_ANNOT}$$

$$\frac{\Gamma \vdash t : \rho}{\Gamma \vdash t : \forall a.\rho}\ \text{TYP\_GEN}$$

$$\frac{\Gamma \vdash t : \forall a.\rho}{\Gamma \vdash t : [a \mapsto \tau]\,\rho}\ \text{TYP\_INST}$$

$\boxed{t \longrightarrow u}$

$$\frac{u \longrightarrow u'}{\mathtt{let}\ x = u\ \mathtt{in}\ t \longrightarrow \mathtt{let}\ x = u'\ \mathtt{in}\ t}\ \text{STEP\_LET}1$$

$$\frac{}{\mathtt{let}\ x = v\ \mathtt{in}\ t \longrightarrow t\,[x \rightsquigarrow v]}\ \text{STEP\_LET}$$

$$\frac{t \longrightarrow t'}{t\ u \longrightarrow t'\ u}\ \text{STEP\_APP}1$$

$$\frac{u \longrightarrow u'}{(\lambda x.t)\ u \longrightarrow (\lambda x.t)\ u'}\ \text{STEP\_APP}2$$

$$\frac{}{(\lambda x.t)\ v \longrightarrow t\,[x \rightsquigarrow v]}\ \text{STEP\_APP}$$

$$\frac{u \longrightarrow u'}{(\lambda(x :: \sigma).t)\ u \longrightarrow (\lambda(x :: \sigma).t)\ u'}\ \text{STEP\_ANNOT\_APP}2$$

$$\frac{}{(\lambda(x :: \sigma).t)\ v \longrightarrow t\,[x \rightsquigarrow v]}\ \text{STEP\_ANNOT\_APP}$$

$$\frac{}{t :: \sigma \longrightarrow t}\ \text{STEP\_ERASE}$$

```
Definition rules:          16 good     0 bad
Definition rule clauses:   31 good     0 bad
```