Caffeine: A JS Profiler

Times and Traces an Object's Method calls

Simple Sample Code

```
HackerSchooler = {
   type: function(){...};
   think: function(){...}
   code: function()
      while(!finished){
         this.think()
         this.type()
      };
   };
};

hs = new HackerSchool()
hs.code()
```

Timing

How long does each function take to run?

```
HackerSchooler = {
   type: function(){
      start = new Date.getTime()
      ...
      end = new Date.getTime()
      "this".runs.push(end-start)
   };
    ...
};
```

==> a list of each function's runtimes

Stack

Which functions call which functions?

```
HackerSchooler.code: function(){
   while(!finished){
      this.think()
      this.type()
   };
};
```

Stack

Which functions call which functions?

```
HackerSchooler.code: function(){
   while(!finished){
      this.think()
      this.type()
   };
};
```

Premise: Put something on the stack when its called, take it off and stash it when it’s done

stack = [ ]

Stack

Which functions call which functions?

```
HackerSchooler.code: function(){
   while(!finished){
      this.think()
      this.type()
   };
};
```

Premise: Put something on the stack when its called, take it off and stash it when it's done

stack = [`code`]

Stack

Which functions call which functions?

```
HackerSchooler.code: function(){
   while(!finished){
      this.think()
      this.type()
   };
};
```

Premise: Put something on the stack when its called, take it off and stash it when it's done

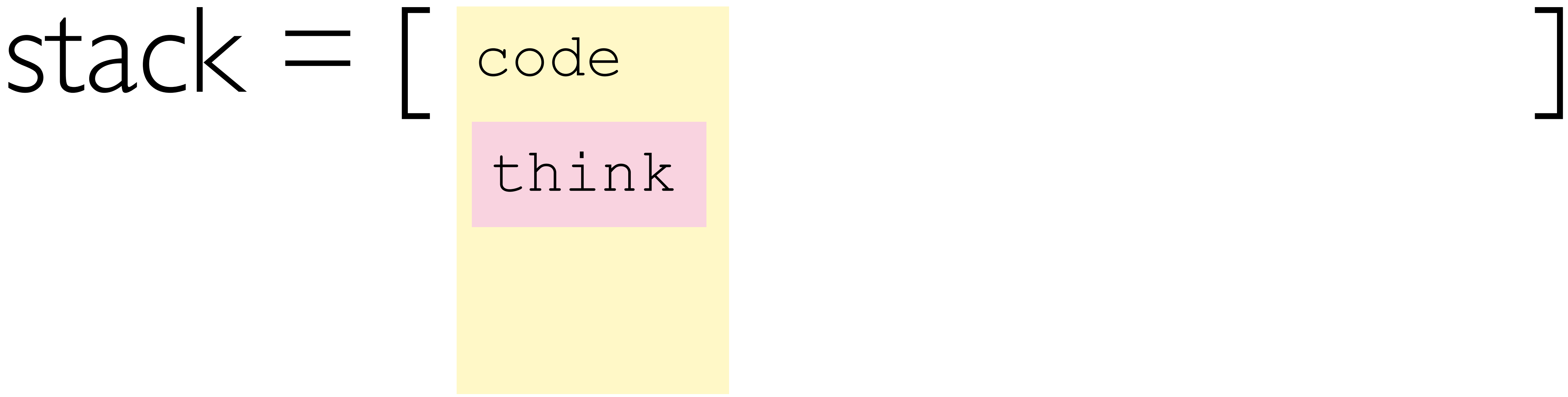
stack = [`code` `think`]

Stack

Which functions call which functions?

```
HackerSchooler.code: function(){
   while(!finished){
      this.think()
      this.type()
   };
};
```

Premise: Put something on the stack when its called, take it off and stash it when it's done



stack = [

code

think

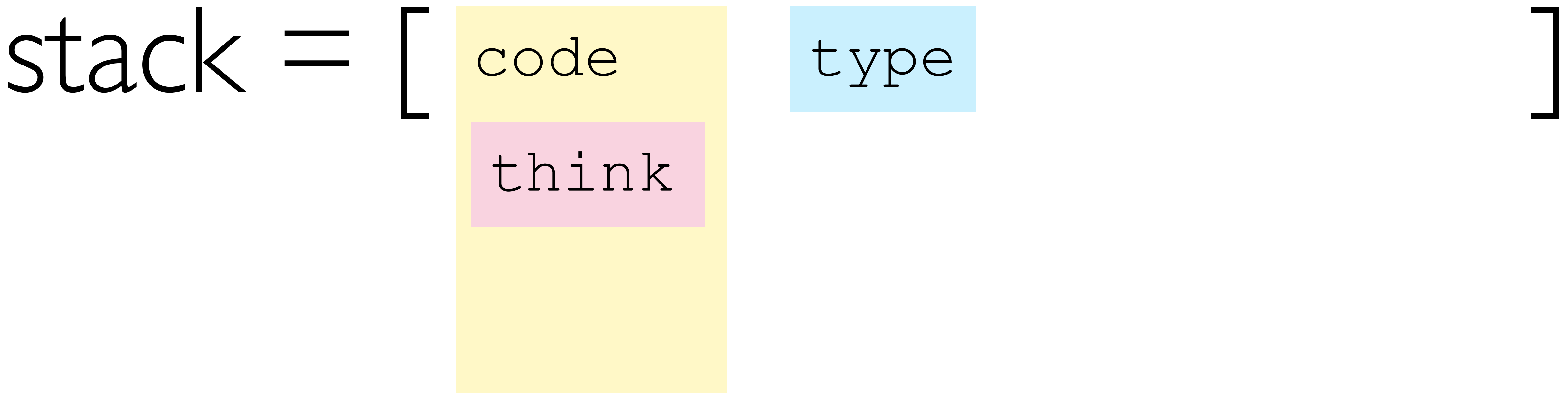
]

Stack

Which functions call which functions?

```
HackerSchooler.code: function(){
   while(!finished){
      this.think()
      this.type()
   };
};
```

Premise: Put something on the stack when its called, take it off and stash it when it's done



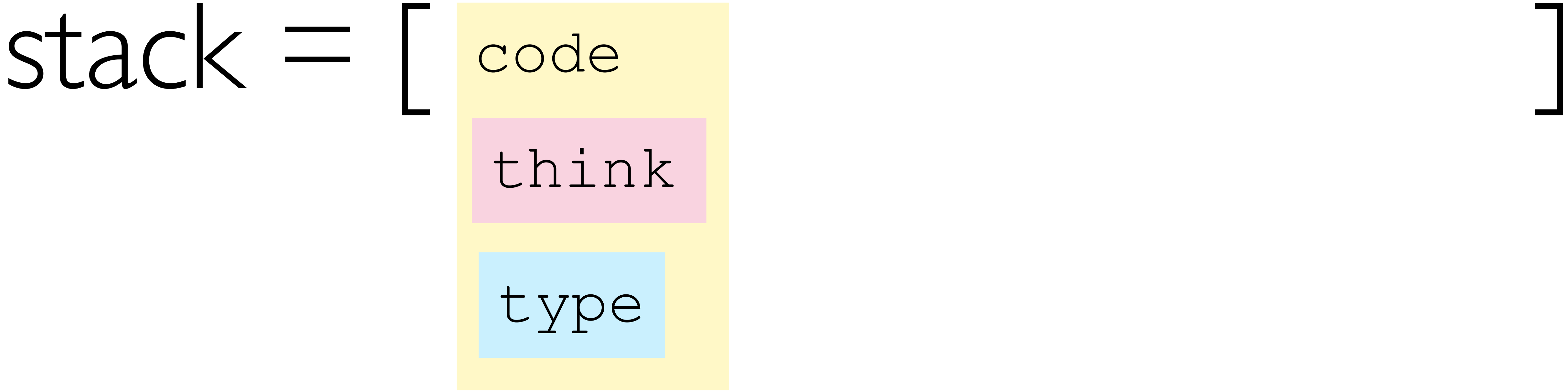
stack = [code think type]

Stack

Which functions call which functions?

```
HackerSchooler.code: function(){
   while(!finished){
      this.think()
      this.type()
   };
};
```

Premise: Put something on the stack when its called, take it off and stash it when it's done



stack = [

code

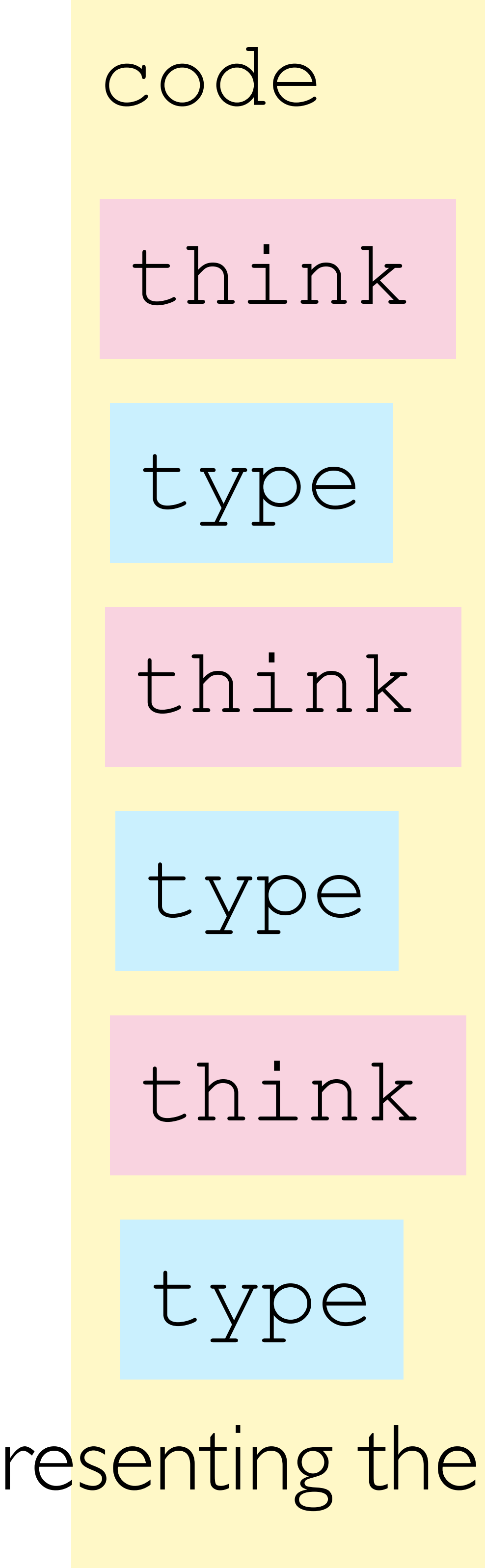
think

type

]

Stack Tree

Which functions call which functions?



code

think

type

think

type

think

type

==> a list of "trees" representing the flow of each top-level method