

# CO<sup>2</sup>DES

INTERACTIVE LIVE CODING EXPERIENCE

**ANTONIO GIGANTI  
CARLO PULVIRENTI  
LORENZO TALONE**

# WHY



---

poor interaction in  
live-coding events

---

scarse interaction in  
streaming events

---

barrier between  
performer and  
audience

# GOALS

The background features a dark, abstract design with a central point from which numerous thin, multi-colored lines radiate outwards, creating a starburst or network effect. The colors include shades of green, blue, red, and yellow, all set against a black background.

—  
enhance the live  
coding experience

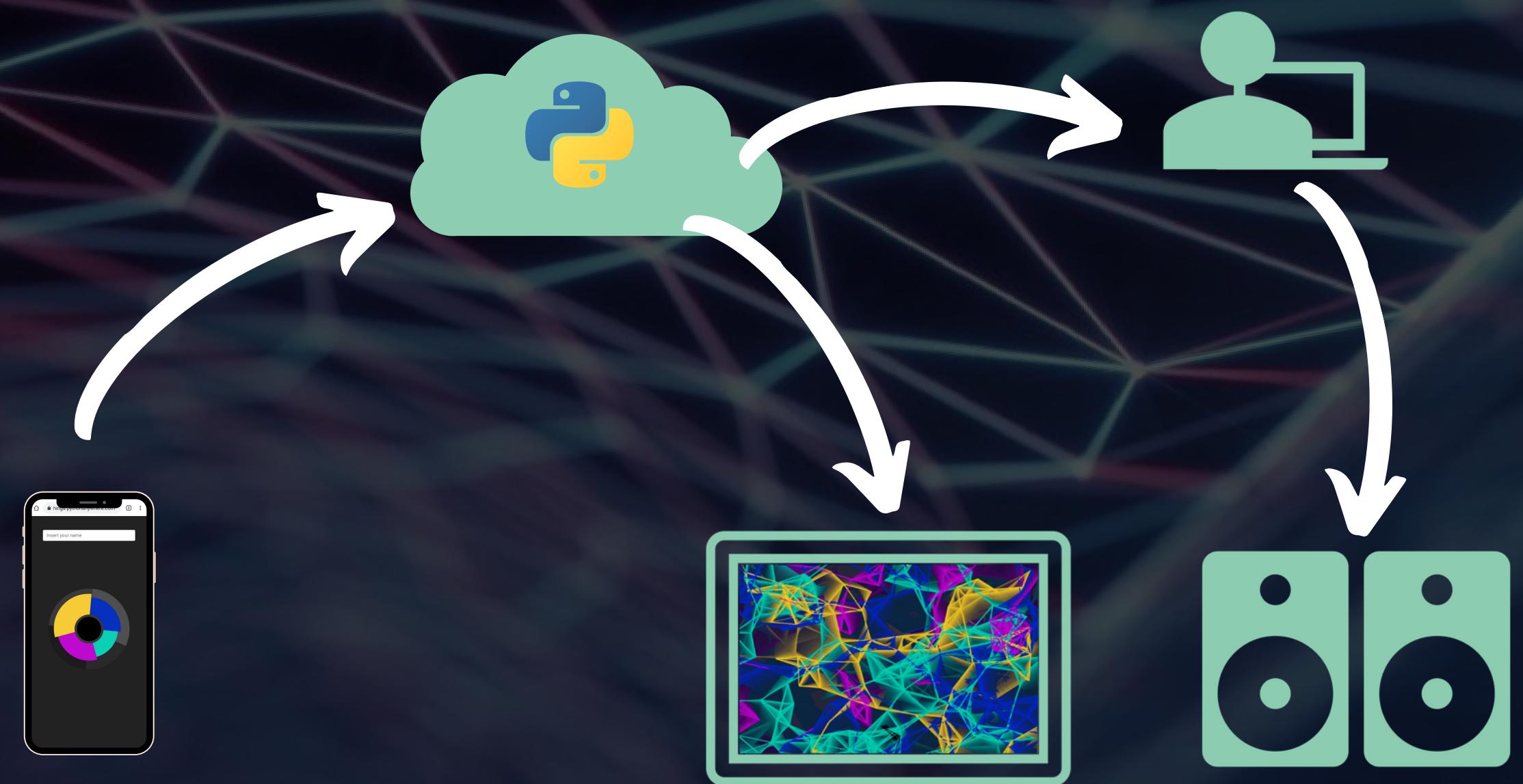
—  
audience interaction

—  
collaborative performance

# INTERACTION

—  
user-machine interaction

—  
audio and visual changes

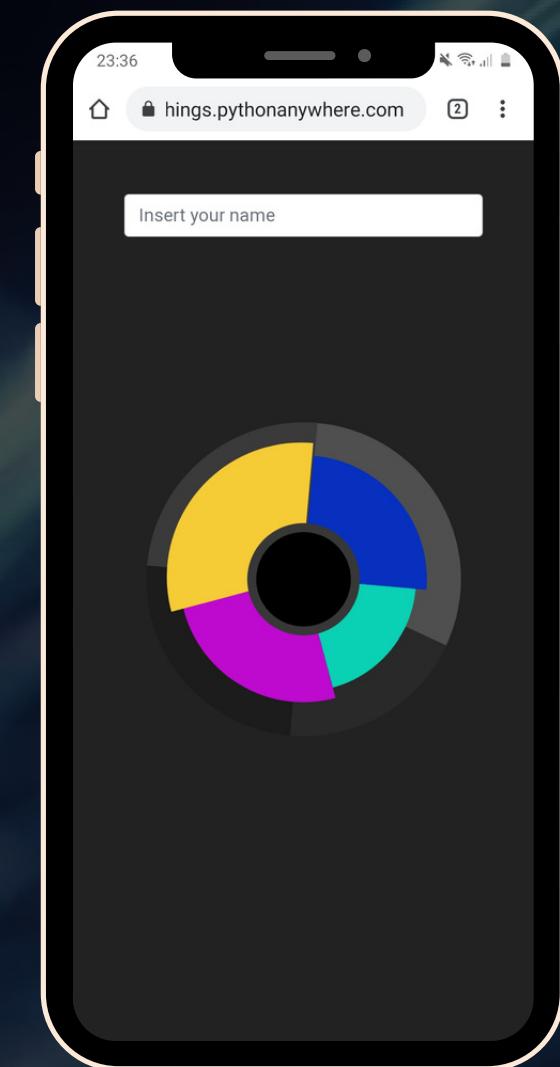


# WEB APP

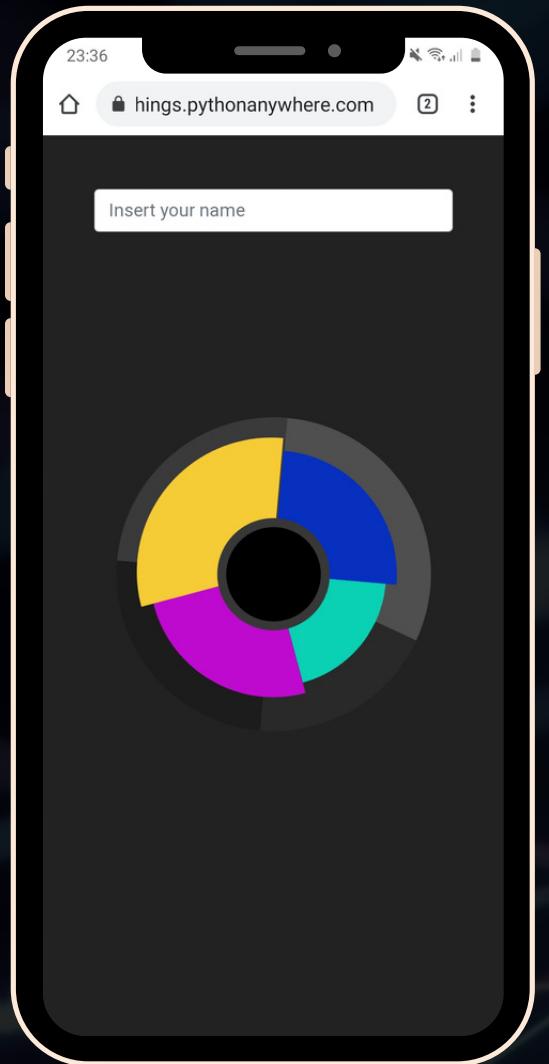


— **color selection**

— **parameter setting**



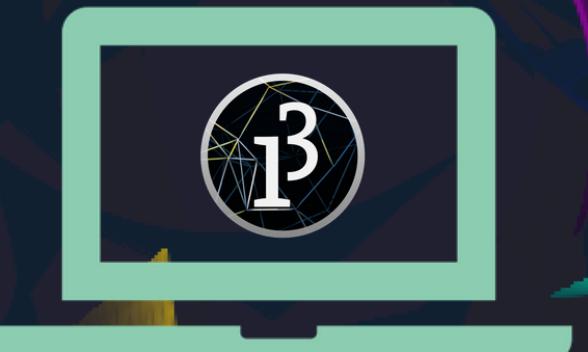
# S E R V E R



- 
- developed in **python**
  - dispatching messages through **HTTP**



# VISUAL



— developed in **Processing**

— every interaction generates geometric elements

— colors and shapes variation

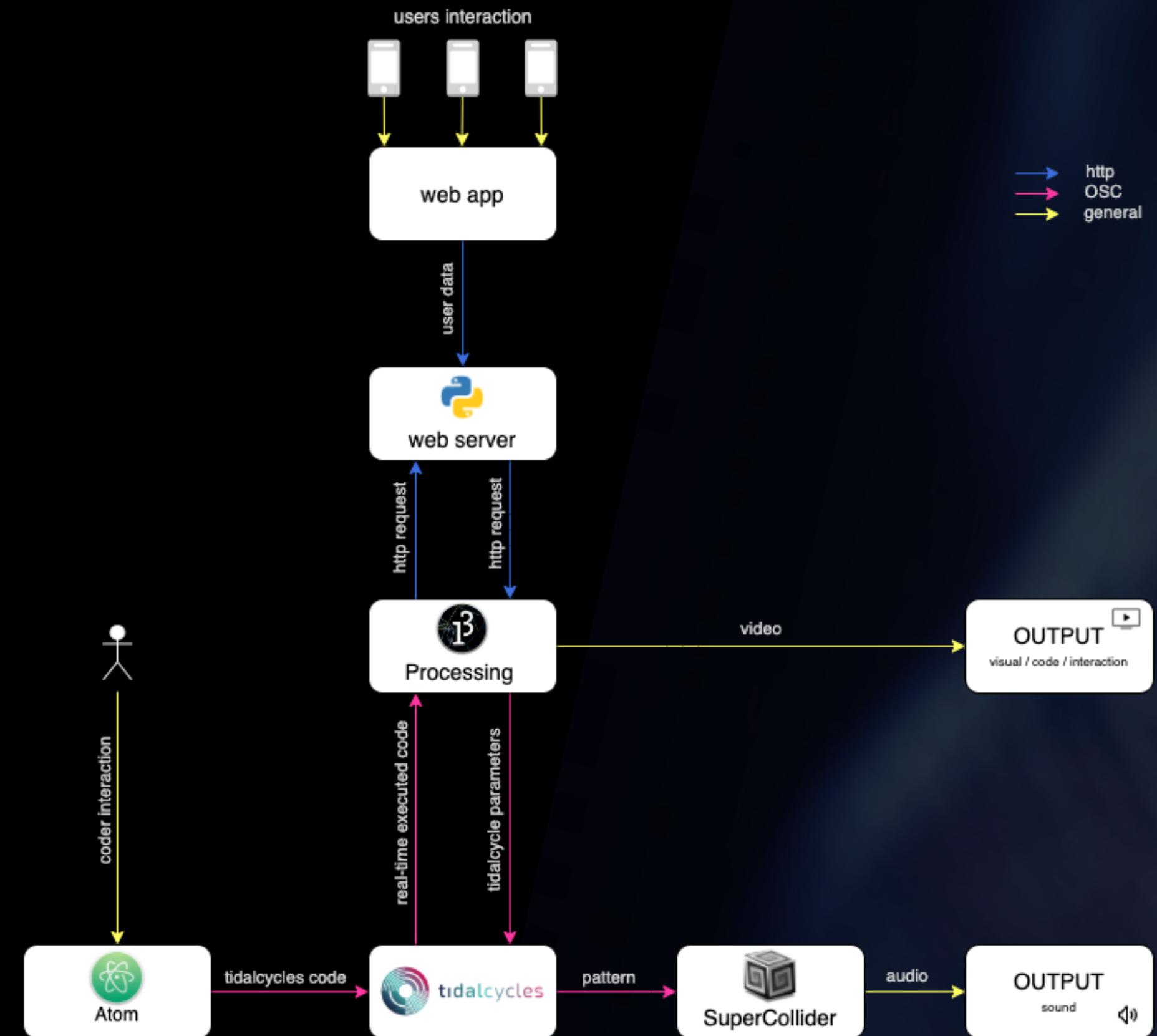
# AUDIO



- 
- interaction between the **coder** and the **audience**
  - live coding done with **TidalCycles**
  - every color controls a different predefined parameter



# FLOWCHART



# FUTURE IMPROVEMENTS

—  
implementation of  
an **evolutionary  
system** to  
generate novel  
and unexpected  
audio and video  
features

—  
giving the audience  
**more parameters** to  
play with

—  
**auto execution:** the  
users' interaction can  
also automatically  
execute some code  
snippets

—  
make the **visual**  
**more responsive** to  
the audio

# LIVE DEMO



```
215 (def move-me
216   (->
217     ((gen-ps (:id pink-squares)) hept)
218     (style {:transform-origin "center" :transform "scale(1.4)"})
219     (anim "woosh-2" "3s" "infinite")
220     (poly)
221     (atom)))
222
223 (def move-me-2
224   (->
225     ((gen-ps (:id pink-squares)) hept)
226     (style {:transform-origin "center" :transform "scale(1.4)"})
227     (anim "woosh" "2s" "infinite")
228     (poly)
229     (atom)))
230
231 (def move-me-3
232   (->
233     ((gen-ps (:id pink-squares)) hept)
234     (gen-sc
235       (style {:transform-origin "center" :transform "scale(1.4)"})
236       (f gen-ss
237         (n-shape
238           (n-ds
239             (list
240               ge
241               offset-line
242               GE
243               RATORS
244               (let [h (se
245                 ings :height)])
246               0.0000000000000002 234.17
247               LF  UTF-8  Closure  (λ)  ↵  master
248               ↵  2 files
249             )
250           )
251         )
252       )
253     )
254   )
255 )
256 )
257 )
258 )
259 )
260 )
261 )
262 )
263 )
264 )
265 )
266 )
267 )
268 )
269 )
270 )
271 )
272 )
273 )
274 )
275 )
276 )
277 )
278 )
279 )
280 )
281 )
282 )
283 )
284 )
285 )
286 )
287 )
288 )
289 )
290 )
291 )
292 )
293 )
294 )
295 )
296 )
297 )
298 )
299 )
299 )
```