

[^ 이전글](#)[^ 다음글](#)[목록](#)[| 기타 질문 | >](#)

Book

**hyun123** 챗봇 마스터 🏆 1:1 채팅

2020.02.21. 21:31 조회 28

💬 댓글 0 URL 복사 ⋮

제가 파이썬 책을 사서 공부 하려 하는데 자스에 도움이 되려나요? 파이썬은 잘 아는데 자스는 잘.. (모르는 건 아니고 알긴 알아요;)

**hyun123**님의 게시글 더보기 >

❤️ 좋아요 0 💬 댓글 0

🔗 공유 | 신고

댓글

댓글알림 ☐**Hibot**

댓글을 남겨보세요



등록

[✎ 글쓰기](#)[답글](#)[목록](#)[▲ TOP](#)

'기타 질문' 게시판 글

이 게시판 새글 구독하기 ☐[\[어플 관련\] 가상 전화번호 \[2\]](#)

hyun123

2020.02.22.

[이용자보호조치 너무 안풀려요 \[6\]](#)

백신

2020.02.22.

Book

hyun123

2020.02.21.

이 카페 인기글

```
Error(재팅수(LTE  
Error: "FS" is not  
#14) at  
):14 (respons
```

채팅순위봇이 갑자기 안되네요..ㅠ

간편 자동응답과 한글코딩을 메인으로 제공
하는 카톡봇앱 개발 예정

성빈

♥ 1 💬 16

```

200  if (isFinite(x) && !isFinite(y)) throw
201    RangeError("Cannot convert NaN to a BigInt");
202  if (isFinite(x) && !isFinite(y))
203    return BigInt(x * y);
204  if (isFinite(x) && !isFinite(y)) {
205    return BigInt(x * y);
206  } else if (isFinite(x) && !isFinite(y)) {
207    return BigInt(x * y);
208  }
209  return
210    NaN;
211 }
212
213 BigInt.prototype.valueOf = function() {
214   return Number(this);
215 }
216
217 BigInt.prototype.toString = function(radix) {
218   return this.toString(radix, 1);
219 }
220
221 BigInt.prototype.toLocaleString = function() {
222   return this.toString();
223 }
224
225 BigInt.prototype.toJSON = function() {
226   return this.toString();
227 }
228
229 BigInt.prototype.toNumber = function() {
230   return Number(this);
231 }
232
233 BigInt.prototype.toBigInt = function() {
234   return this;
235 }
236
237 BigInt.prototype.toInteger = function() {
238   return this;
239 }
240
241 BigInt.prototype.toFloat = function() {
242   return Number(this);
243 }
244
245 BigInt.prototype.toInteger = function() {
246   return this;
247 }
248
249 BigInt.prototype.toFloat = function() {
250   return Number(this);
251 }
252
253 BigInt.prototype.toInteger = function() {
254   return this;
255 }
256
257 BigInt.prototype.toFloat = function() {
258   return Number(this);
259 }
260
261 BigInt.prototype.toInteger = function() {
262   return this;
263 }
264
265 BigInt.prototype.toFloat = function() {
266   return Number(this);
267 }
268
269 BigInt.prototype.toInteger = function() {
270   return this;
271 }
272
273 BigInt.prototype.toFloat = function() {
274   return Number(this);
275 }
276
277 BigInt.prototype.toInteger = function() {
278   return this;
279 }
280
281 BigInt.prototype.toFloat = function() {
282   return Number(this);
283 }
284
285 BigInt.prototype.toInteger = function() {
286   return this;
287 }
288
289 BigInt.prototype.toFloat = function() {
290   return Number(this);
291 }
292
293 BigInt.prototype.toInteger = function() {
294   return this;
295 }
296
297 BigInt.prototype.toFloat = function() {
298   return Number(this);
299 }
300
301 BigInt.prototype.toInteger = function() {
302   return this;
303 }
304
305 BigInt.prototype.toFloat = function() {
306   return Number(this);
307 }
308
309 BigInt.prototype.toInteger = function() {
310   return this;
311 }
312
313 BigInt.prototype.toFloat = function() {
314   return Number(this);
315 }
316
317 BigInt.prototype.toInteger = function() {
318   return this;
319 }
320
321 BigInt.prototype.toFloat = function() {
322   return Number(this);
323 }
324
325 BigInt.prototype.toInteger = function() {
326   return this;
327 }
328
329 BigInt.prototype.toFloat = function() {
330   return Number(this);
331 }
332
333 BigInt.prototype.toInteger = function() {
334   return this;
335 }
336
337 BigInt.prototype.toFloat = function() {
338   return Number(this);
339 }
340
341 BigInt.prototype.toInteger = function() {
342   return this;
343 }
344
345 BigInt.prototype.toFloat = function() {
346   return Number(this);
347 }
348
349 BigInt.prototype.toInteger = function() {
350   return this;
351 }
352
353 BigInt.prototype.toFloat = function() {
354   return Number(this);
355 }
356
357 BigInt.prototype.toInteger = function() {
358   return this;
359 }
360
361 BigInt.prototype.toFloat = function() {
362   return Number(this);
363 }
364
365 BigInt.prototype.toInteger = function() {
366   return this;
367 }
368
369 BigInt.prototype.toFloat = function() {
370   return Number(this);
371 }
372
373 BigInt.prototype.toInteger = function() {
374   return this;
375 }
376
377 BigInt.prototype.toFloat = function() {
378   return Number(this);
379 }
380
381 BigInt.prototype.toInteger = function() {
382   return this;
383 }
384
385 BigInt.prototype.toFloat = function() {
386   return Number(this);
387 }
388
389 BigInt.prototype.toInteger = function() {
390   return this;
391 }
392
393 BigInt.prototype.toFloat = function() {
394   return Number(this);
395 }
396
397 BigInt.prototype.toInteger = function() {
398   return this;
399 }
400
401 BigInt.prototype.toFloat = function() {
402   return Number(this);
403 }
404
405 BigInt.prototype.toInteger = function() {
406   return this;
407 }
408
409 BigInt.prototype.toFloat = function() {
410   return Number(this);
411 }
412
413 BigInt.prototype.toInteger = function() {
414   return this;
415 }
416
417 BigInt.prototype.toFloat = function() {
418   return Number(this);
419 }
420
421 BigInt.prototype.toInteger = function() {
422   return this;
423 }
424
425 BigInt.prototype.toFloat = function() {
426   return Number(this);
427 }
428
429 BigInt.prototype.toInteger = function() {
430   return this;
431 }
432
433 BigInt.prototype.toFloat = function() {
434   return Number(this);
435 }
436
437 BigInt.prototype.toInteger = function() {
438   return this;
439 }
440
441 BigInt.prototype.toFloat = function() {
442   return Number(this);
443 }
444
445 BigInt.prototype.toInteger = function() {
446   return this;
447 }
448
449 BigInt.prototype.toFloat = function() {
450   return Number(this);
451 }
452
453 BigInt.prototype.toInteger = function() {
454   return this;
455 }
456
457 BigInt.prototype.toFloat = function() {
458   return Number(this);
459 }
460
461 BigInt.prototype.toInteger = function() {
462   return this;
463 }
464
465 BigInt.prototype.toFloat = function() {
466   return Number(this);
467 }
468
469 BigInt.prototype.toInteger = function() {
470   return this;
471 }
472
473 BigInt.prototype.toFloat = function() {
474   return Number(this);
475 }
476
477 BigInt.prototype.toInteger = function() {
478   return this;
479 }
480
481 BigInt.prototype.toFloat = function() {
482   return Number(this);
483 }
484
485 BigInt.prototype.toInteger = function() {
486   return this;
487 }
488
489 BigInt.prototype.toFloat = function() {
490   return Number(this);
491 }
492
493 BigInt.prototype.toInteger = function() {
494   return this;
495 }
496
497 BigInt.prototype.toFloat = function() {
498   return Number(this);
499 }
500
501 BigInt.prototype.toInteger = function() {
502   return this;
503 }
504
505 BigInt.prototype.toFloat = function() {
506   return Number(this);
507 }
508
509 BigInt.prototype.toInteger = function() {
510   return this;
511 }
512
513 BigInt.prototype.toFloat = function() {
514   return Number(this);
515 }
516
517 BigInt.prototype.toInteger = function() {
518   return this;
519 }
520
521 BigInt.prototype.toFloat = function() {
522   return Number(this);
523 }
524
525 BigInt.prototype.toInteger = function() {
526   return this;
527 }
528
529 BigInt.prototype.toFloat = function() {
530   return Number(this);
531 }
532
533 BigInt.prototype.toInteger = function() {
534   return this;
535 }
536
537 BigInt.prototype.toFloat = function() {
538   return Number(this);
539 }
540
541 BigInt.prototype.toInteger = function() {
542   return this;
543 }
544
545 BigInt.prototype.toFloat = function() {
546   return Number(this);
547 }
548
549 BigInt.prototype.toInteger = function() {
550   return this;
551 }
552
553 BigInt.prototype.toFloat = function() {
554   return Number(this);
555 }
556
557 BigInt.prototype.toInteger = function() {
558   return this;
559 }
560
561 BigInt.prototype.toFloat = function() {
562   return Number(this);
563 }
564
565 BigInt.prototype.toInteger = function() {
566   return this;
567 }
568
569 BigInt.prototype.toFloat = function() {
570   return Number(this);
571 }
572
573 BigInt.prototype.toInteger = function() {
574   return this;
575 }
576
577 BigInt.prototype.toFloat = function() {
578   return Number(this);
579 }
580
581 BigInt.prototype.toInteger = function() {
582   return this;
583 }
584
585 BigInt.prototype.toFloat = function() {
586   return Number(this);
587 }
588
589 BigInt.prototype.toInteger = function() {
590   return this;
591 }
592
593 BigInt.prototype.toFloat = function() {
594   return Number(this);
595 }
596
597 BigInt.prototype.toInteger = function() {
598   return this;
599 }
600
601 BigInt.prototype.toFloat = function() {
602   return Number(this);
603 }
604
605 BigInt.prototype.toInteger = function() {
606   return this;
607 }
608
609 BigInt.prototype.toFloat = function() {
610   return Number(this);
611 }
612
613 BigInt.prototype.toInteger = function() {
614   return this;
615 }
616
617 BigInt.prototype.toFloat = function() {
618   return Number(this);
619 }
620
621 BigInt.prototype.toInteger = function() {
622   return this;
623 }
624
625 BigInt.prototype.toFloat = function() {
626   return Number(this);
627 }
628
629 BigInt.prototype.toInteger = function() {
630   return this;
631 }
632
633 BigInt.prototype.toFloat = function() {
634   return Number(this);
635 }
636
637 BigInt.prototype.toInteger = function() {
638   return this;
639 }
640
641 BigInt.prototype.toFloat = function() {
642   return Number(this);
643 }
644
645 BigInt.prototype.toInteger = function() {
646   return this;
647 }
648
649 BigInt.prototype.toFloat = function() {
650   return Number(this);
651 }
652
653 BigInt.prototype.toInteger = function() {
654   return this;
655 }
656
657 BigInt.prototype.toFloat = function() {
658   return Number(this);
659 }
660
661 BigInt.prototype.toInteger = function() {
662   return this;
663 }
664
665 BigInt.prototype.toFloat = function() {
666   return Number(this);
667 }
668
669 BigInt.prototype.toInteger = function() {
670   return this;
671 }
672
673 BigInt.prototype.toFloat = function() {
674   return Number(this);
675 }
676
677 BigInt.prototype.toInteger = function() {
678   return this;
679 }
680
681 BigInt.prototype.toFloat = function() {
682   return Number(this);
683 }
684
685 BigInt.prototype.toInteger = function() {
686   return this;
687 }
688
689 BigInt.prototype.toFloat = function() {
690   return Number(this);
691 }
692
693 BigInt.prototype.toInteger = function() {
694   return this;
695 }
696
697 BigInt.prototype.toFloat = function() {
698   return Number(this);
699 }
700
701 BigInt.prototype.toInteger = function() {
702   return this;
703 }
704
705 BigInt.prototype.toFloat = function() {
706   return Number(this);
707 }
708
709 BigInt.prototype.toInteger = function() {
710   return this;
711 }
712
713 BigInt.prototype.toFloat = function() {
714   return Number(this);
715 }
716
717 BigInt.prototype.toInteger = function() {
718   return this;
719 }
720
721 BigInt.prototype.toFloat = function() {
722   return Number(this);
723 }
724
725 BigInt.prototype.toInteger = function() {
726   return this;
727 }
728
729 BigInt.prototype.toFloat = function() {
730   return Number(this);
731 }
732
733 BigInt.prototype.toInteger = function() {
734   return this;
735 }
736
737 BigInt.prototype.toFloat = function() {
738   return Number(this);
739 }
740
741 BigInt.prototype.toInteger = function() {
742   return this;
743 }
744
745 BigInt.prototype.toFloat = function() {
746   return Number(this);
747 }
748
749 BigInt.prototype.toInteger = function() {
750   return this;
751 }
752
753
```

틱택토 (Player vs Bot)

[카카오톡 봇] 양산형 자동학습 봇 소스 공유

"시초가 중국발" 발언 관련 해명글 및 부탁
 의 글
 eval 질문
 트윅치

Kim 5 3
0 5

안녕하세요

KHP

0 6

글자의 개수가 넘어가면 자르기 질문

흑까마귀

0 5

```

25  if (parent instanceof Map) {
26      replace.replace(2, 0, new Node( ));
27      return;
28  }
29  }
30  }
31  }
32  }
33  }
34  }
35  }
36  }
37  }
38  }
39  }
40  }

```

혹시 여기서 변수를 저장하는 부분이 어딘지