

[| 일반 소스 공유 | >](#)

[미니게임] 미니게임 종합세트 (퀴즈, 베라, 369, 타자게임)



도미 doami2005 챗봇 창시자 🌞 1:1 채팅

2020.05.27. 19:13 조회 698

💬 댓글 13 🔖 URL 복사 ⋮

📄 첨부파일 1

```
1  LI = "-".repeat(40)
2  Lw = "\u200b".repeat(500)
3  FS = FileStream
4  Jsoup = org.jsoup.Jsoup
5  rooms = ["미니게임"]
6  doami = {
7    "hash": [-354374256, 626170442, -438224563]
8  }
9
10 parse = Jsoup.connect("https://danaw.tistory.com/1")
11 sentences = Array(73).fill().map((e, i) => parse.get(i))
12
13 FastType = {
14   "Path": "sdcard/FastType/scores"
15 }
16
17 FS.read(FastType.Path) || FS.write(FastType.Path, "{
18
19 Quiz = {
20   "Path": "sdcard/quiz/scores",
21   "Prefix": "/",
22   "Json": {}
23 }
24
25 BR = {
26   "Limit": 31,
27   "Prefix": "/베라 ",
28   "Json": {}
29 }
```

```

30
31 TSN = {
32   "Prefix": "/369 ",
33   "Json": {}
34 }
35
36 shuffle = arr => {
37   arr2 = arr.slice(0)
38   for (i = arr2.length; i; i--) {
39     j = Math.random() * i | 0
40     x = arr2[i - 1]
41     arr2[i - 1] = arr2[j]
42     arr2[j] = x
43   }
44   return arr2
45 }
46
47 function response (room, msg, sender, igc, replier, id
48
49   if (rooms.length && rooms.indexOf(room) == -1) re
50
51   if (msg.startsWith(Quiz.Prefix)) {
52     input = msg.substr(Quiz.Prefix.length).trim()
53     if (input == "퀴즈 도움말") {
54       replier.reply([
55         "• 퀴즈 도움말", LI, [
56         "퀴즈",
57         "힌트",
58         "정답",
59         "포기",
60         "점수판"
61       ].map(e => Quiz.Prefix+e).join("\n"), LI].join("\n"))
62     } else if (input == "퀴즈" && !Quiz.Json[room]) {
63       parse = JSON.parse(org.jsoup.Jsoup.connect("h
64       Quiz.Json[room] = [parse.quiz, parse.hint, parse.
65       replier.reply([

```

```

66     "• "+Quiz.Json[room][0], Ll,
67     [
68     "정답 (정답)",
69     "힌트",
70     "포기"
71     ].map(e => Quiz.Prefix+e).join("\n"), Ll
72     ].join("\n"))
73 } else if (input == "힌트" && Quiz.Json[room]) {
74     replier.reply("• 힌트 : "+Quiz.Json[room][1])
75 } else if (input == "포기" && Quiz.Json[room]) {
76     replier.reply([
77     "• 퀴즈를 포기하셨습니다!",
78     "답 : "+Quiz.Json[room][2],
79     "풀이 : "+Quiz.Json[room][3]
80     ].join("\n\n"))
81     delete Quiz.Json[room]
82 } else if (input.startsWith("정답") && Quiz.Json[room]) {
83     input = msg.substr(2+Quiz.Prefix.length).trim()
84     if (input.replace(/ /g, "") == Quiz.Json[room][2].replace(/ /g, "")) {
85         replier.reply([
86         "• "+sender+" 님 정답!",
87         "답 : "+Quiz.Json[room][2],
88         "풀이 : "+Quiz.Json[room][3]
89         ].join("\n\n"))
90         delete Quiz.Json[room]
91         scores = JSON.parse(FS.read(Quiz.Path))
92         scores[room] = scores[room] || {}
93         scores[room][sender] = ++scores[room][sender]
94         FS.write(Quiz.Path, JSON.stringify(scores, null, 2))
95     } else replier.reply("• 틀렸습니다!")
96 } else if (input == "점수판") {
97     scores = JSON.parse(FS.read(Quiz.Path))
98     scores[room] = scores[room] || {}
99     result = Object.keys(scores[room]).sort((a, b) =>
100     replier.reply(result.length?"• 점수판 입니다."+Lw, L
101     } else if (input == "베라 초기화") {

```



```

102     if (!doami.hash.length||(doami.hash.length && dc
103         scores = JSON.parse(FS.read(Quiz.Path))
104         scores[room] = {}
105         FS.write(Quiz.Path, JSON.stringify(scores, null,
106             replier.reply("• 현재 방의 점수판이 초기화 되었습니다
107     })
108 }
109 }
110 if (msg.startsWith(BR.Prefix)) {
111     input = msg.substr(BR.Prefix.length).trim()
112     BR.Json[room] = BR.Json[room] || {
113         "start": false,
114         "players": [],
115         "orders": [],
116         "turn": null,
117         "total": 0
118     }
119     if (input == "도움말") {
120         replier.reply([
121             "• 베라 도움말", LI, [
122                 "생성",
123                 "입장",
124                 "퇴장",
125                 "시작",
126                 "종료"
127             ].map(e => BR.Prefix+e).join("\n"), LI].join("\n"))
128     } else if (input == "생성" || input == "만들기") {
129         if (!BR.Json[room]["players"].length) {
130             BR.Json[room]["players"].push(sender)
131             replier.reply([
132                 "• "+sender+" 님이 방을 생성하셨습니다!", LI,
133                 "1. "+sender, LI
134             ].join("\n"))
135         } else replier.reply("• 이미 생성된 방이 있습니다!")
136     } else if (input == "입장" || input == "참가") {
137         if (BR.Json[room]["players"].length) {

```

```

138     if (!BR.Json[room]["start"]) {
139         if (BR.Json[room]["players"].indexOf(sender) == -1) {
140             BR.Json[room]["players"].push(sender)
141             replier.reply([
142                 "• 방에 입장하셨습니다.", LI,
143                 BR.Json[room]["players"].map((e, i) => (i+1)+
144                     "번").join("\n")
145             ]) } else replier.reply("• 이미 방에 입장하셨습니다!")
146     } else replier.reply("• 게임이 이미 시작되었습니다!")
147     } else replier.reply("• 생성된 방이 없습니다!")
148 } else if (input == "퇴장" || input == "나가기") {
149     if (BR.Json[room]["players"].length) {
150         if (BR.Json[room]["players"].indexOf(sender)+1) {
151             if (!BR.Json[room]["start"]) {
152                 BR.Json[room]["players"].splice(BR.Json[room]
153                     ["players"].indexOf(sender), 1)
154                 if (BR.Json[room]["players"].length) {
155                     replier.reply([
156                         "• "+sender+" 님이 방에서 퇴장하셨습니다.", LI,
157                         BR.Json[room]["players"].map((e, i) => (i+1)+
158                             "번").join("\n")
159                     ]) } else {
160                         delete BR.Json[room]
161                         replier.reply([
162                             "• "+sender+" 님이 방에서 퇴장하셨습니다.",
163                             "방의 인원이 0명이 되어 방이 삭제됩니다!"
164                         ]).join("\n")
165                     } } else replier.reply("• 게임 도중엔 나가실 수 없습니
166     ") } else replier.reply("• 방에 입장하지 않으셨습니다!")
167     } else replier.reply("• 생성된 방이 없습니다!")
168 } else if (input == "시작") {
169     if (BR.Json[room]["players"].length) {
170         if (!BR.Json[room]["start"]) {
171             if (BR.Json[room]["players"][0] == sender) {
172                 if (BR.Json[room]["players"].length > 1) {
173                     BR.Json[room]["orders"] = shuffle(BR.Json[ro

```



```

174     replier.reply([
175         "• 게임이 시작되었습니다!\n",
176         "[ 순서 ]",
177         BR.Json[room]["orders"].map((e, i) => (i+1)+"
178     ]).join("\n"))
179     replier.reply(BR.Json[room]["orders"][0]+" 님은
180     BR.Json[room]["turn"] = BR.Json[room]["orde
181     } else replier.reply("• 2명 이상만 게임을 시작할 수
182     } else replier.reply("• 방장만 게임을 시작할 수 있습
183     } else replier.reply("• 게임이 이미 시작되었습니다!")
184     } else replier.reply("• 생성된 방이 없습니다!")
185 } else if (input == "종료") {
186     if (BR.Json[room]["start"]) {
187         if (BR.Json[room]["players"].indexOf(sender)+1)
188         if (BR.Json[room]["players"][0] == sender) {
189             delete BR.Json[room]
190             replier.reply("• 게임이 종료되었습니다.")
191         } else replier.reply("• 방장만 게임을 종료할 수 있습
192         } else replier.reply("• 게임에 참여하지 않으셨습니다
193         } else replier.reply("• 아직 게임이 시작되지 않았습니
194     } else replier.reply("• 없는 명령어입니다!")
195 }
196 if (!isNaN(msg) && BR.Json[room] && BR.Json[roo
197     if (Array(3).fill().map((i, i) => BR.Json[room]["total"
198         BR.Json[room]["total"] = Number(msg)
199         if (BR.Json[room]["total"]+1 > BR.Limit) {
200             BR.Json[room]["players"].splice(BR.Json[room]
201             replier.reply("• 이런, "+sender+" 님이 탈락하였습니
202             delete BR.Json[room]
203         } else {
204             BR.Json[room]["turn"] = BR.Json[room]["orders"]
205             replier.reply("• "+BR.Json[room]["total"]+"\n"+BR
206         }
207     } else replier.reply("• 숫자는 "+Array(3).fill().map((i,
208 }
209 if (msg.startsWith(TSN.Prefix)) {

```

```

210 input = msg.substr(TSN.Prefix.length).trim()
211 TSN.Json[room] = TSN.Json[room] || {
212   "start": false,
213   "players": [],
214   "orders": [],
215   "turn": null,
216   "current": 0
217 }
218 if (input == "도움말") {
219   replier.reply([
220     "• 369 도움말", LI, [
221       "생성",
222       "입장",
223       "퇴장",
224       "시작",
225       "종료"
226     ].map(e => TSN.Prefix+e).join("\n"), LI].join("\n"))
227 } else if (input == "생성" || input == "만들기") {
228   if (!TSN.Json[room]["players"].length) {
229     TSN.Json[room]["players"].push(sender)
230     replier.reply([
231       "• "+sender+" 님이 방을 생성하셨습니다!", LI,
232       "1. "+sender, LI
233     ].join("\n"))
234   } else replier.reply("• 이미 생성된 방이 있습니다!")
235 } else if (input == "입장" || input == "참가") {
236   if (TSN.Json[room]["players"].length) {
237     if (!TSN.Json[room]["start"]) {
238       if (TSN.Json[room]["players"].indexOf(sender)
239         TSN.Json[room]["players"].push(sender)
240         replier.reply([
241           "• 방에 입장하셨습니다.", LI,
242           TSN.Json[room]["players"].map((e, i) => (i+1)+
243             ].join("\n"))
244         } else replier.reply("• 이미 방에 입장하셨습니다!")
245         } else replier.reply("• 게임이 이미 시작되었습니다!")

```



```

246 } else replier.reply("• 생성된 방이 없습니다!")
247 } else if (input == "퇴장" || input == "나가기") {
248   with (TSN.Json[room]) {
249     if (players.length) {
250       if (players.indexOf(sender)+1) {
251         if (!start) {
252           players.splice(players.indexOf(sender), 1)
253           if (players.length) {
254             replier.reply([
255               "• "+sender+" 님이 방에서 퇴장하셨습니다.", LI,
256               players.map((e, i) => (i+1)+" ".+e).join("\n"),
257               ].join("\n"))
258           } else {
259             delete TSN.Json[room]
260             replier.reply([
261               "• "+sender+" 님이 방에서 퇴장하셨습니다.",
262               "방의 인원이 0명이 되어 방이 삭제됩니다!"
263               ].join("\n"))
264           }
265         } else replier.reply("• 게임 도중엔 나가실 수 없습니다!")
266       } else replier.reply("• 방에 입장하지 않으셨습니다!")
267     } else replier.reply("• 생성된 방이 없습니다!")
268   }
269 } else if (input == "시작") {
270   with (TSN.Json[room]) {
271     if (players.length) {
272       if (!start) {
273         if (players[0] == sender) {
274           if (players.length > 1) {
275             orders = shuffle(players)
276             replier.reply([
277               "• 게임이 시작되었습니다!\n",
278               "[ 순서 ]",
279               orders.map((e, i) => (i+1)+" ".+e).join("\n")
280               ].join("\n"))
281             replier.reply(orders[0]+" 님부터 시작해 주세요

```



```

282         turn = orders[0]
283         return;
284     } else replier.reply("• 2명 이상만 게임을 시작할 수 있습니다!")
285     } else replier.reply("• 방장만 게임을 시작할 수 있습니다!")
286     } else replier.reply("• 게임이 이미 시작되었습니다!")
287     } else replier.reply("• 생성된 방이 없습니다!")
288 }
289 } else if (input == "종료") {
290     with (TSN.Json[room]) {
291         if (start) {
292             if (players.indexOf(sender)+1) {
293                 if (players[0] == sender) {
294                     delete TSN.Json[room]
295                     replier.reply("• 게임이 종료되었습니다.")
296                 } else replier.reply("• 방장만 게임을 종료할 수 있습니다!")
297                 } else replier.reply("• 게임에 참여하지 않으셨습니다!")
298                 } else replier.reply("• 아직 게임이 시작되지 않았습니다!")
299             }
300         } else replier.reply("• 없는 명령어입니다!")
301     }
302     if (TSN.Json[room] && TSN.Json[room]["turn"] == sender) {
303         with (TSN.Json[room]) {
304             current++
305             turn = orders[((orders.indexOf(turn)+1)<(orders.length))%orders.length]
306             if ((["3", "6", "9"].some(e => String(current).includes(e)))) {
307                 replier.reply("• "+turn+" 님의 차례입니다.")
308             } else {
309                 players.splice(players.indexOf(sender), 1)
310                 orders.splice(orders.indexOf(sender), 1)
311                 replier.reply([
312                     "• 이런, "+sender+" 님이 탈락하였습니다!",
313                     "[ 생존자 ]\n"+players.map((e, i) => (i+1)+"번 "+e).join("\n\n"))
314                 if (players.length != 1) {
315                     replier.reply("• "+turn+" 님은 "+current+" 부터 시작합니다!")
316                 } else {
317                     replier.reply("• 게임이 종료되었습니다!")
318                 }
319             }
320         }
321     }
322 }

```

```

318     replier.reply([
319         "• 인원이 1명이 되어 게임이 종료됩니다.",
320         "승리자는 " + players[0] + " 님 입니다!"
321     ].join("\n"))
322     delete TSN.Json[room]
323 }
324 }
325 }
326 }
327 if (msg == "/타자 도움말") {
328     replier.reply([
329         "• 타자 게임 도움말 ", LI,
330         "/타자 게임",
331         "/기록판", LI
332     ].join("\n"))
333 }
334 if (msg == "/타자 게임") {
335     if (!FastType.player) {
336         replier.reply([
337             "• 타자 게임을 시작합니다!",
338             "빠른 시간 내에 문장을 입력해 주세요.",
339             "확인' 이라 입력 시 스톱워치가 시작되며 문장이 출력됨
340         ].join("\n"))
341         FastType.player = sender
342     } else replier.reply("• 이미 누군가가 게임을 플레이 중임
343 }
344 if (sender == FastType.player && msg == "확인" && !
345     FastType.start ^= 1
346     st = new Date().getTime()
347     FastType.sentence = sentences[Math.random()*s
348     replier.reply([
349         "• 타이머가 시작되었습니다!",
350         "빠른 시간 내에 문장을 입력해 주세요.",
351         "[" + FastType.sentence + "]"
352     ].join("\n"))
353     return;
354 }

```



```

354 }
355 if (FastType.start && sender == FastType.player) {
356   if (msg == FastType.sentence.replace(/u200b/g,
357     end = new Date().getTime()
358     replier.reply([
359       "• "+sender+" 님 성공!",
360       "걸린 시간 : "+((end-st)/1000)+"초"
361     ].join("\n"))
362     json = JSON.parse(FS.read(FastType.Path))
363     json[sender] = json[sender] || []
364     json[sender].push((end - st) / 1000)
365     FS.write(FastType.Path, JSON.stringify(json, null
366   } else replier.reply("• 도전 실패입니다! 나중에 다시 시!
367   FastType = {
368     "Path" : "sdcard/FastType/scores"
369   }
370 }
371 if (msg == "/기록판") {
372   json = JSON.parse(FS.read(FastType.Path))
373   result = Object.keys(json).map(e => e+"\n"+json[e]
374   replier.reply("• 타자 게임 기록판"+Lw+"\n\n"+result)
375 }
376
377 }

```

다 합쳤습니다 !



도미 doami2005님의 게시글 더보기 >

❤ 좋아요 11 💬 댓글 13

🔗 공유 | 신고

댓글 등록순 최신순 C

댓글알림 ☐



하린
소스의 길이가 길드....

2020.05.27. 19:14 답글 쓰기



재승
와 종합 선물 세트

2020.05.27. 19:15 답글 쓰기



deadmau5

ㄷ ㄷ ㄷ ㄷ

2020.05.27. 19:16 답글 쓰기



Devil1234

ㅎ ㄷ ㄷ

2020.05.27. 19:28 답글 쓰기



Devil1234

return은 뭔가요?

2020.05.27. 19:29 답글 쓰기



인류역사상최악의사건

함수 종료 및 값 반환

2020.05.27. 19:29 답글 쓰기



Devil1234

인류역사상최악의사건 냉

2020.05.27. 19:31 답글 쓰기



엄준식

와우 역시 도미님 감사합니다

2020.05.27. 19:35 답글 쓰기



복숭아사춘기

감사합니다

2020.05.27. 20:04 답글 쓰기



백작선생

빨리 내리면 꿈틀거림

2020.05.27. 20:57 답글 쓰기



와 샌즈

나

2020.05.28. 07:47 답글 쓰기



봇초보입니당

어케 쓰나요 사용법좀

2020.06.22. 13:59 답글 쓰기



0주주인

정말 죄송하지만... 혹시 제가 따로 수정해야하는 부분이 있나요? ㅠㅠㅠㅠ

2020.06.23. 06:47 답글 쓰기

Hibot

댓글을 남겨보세요



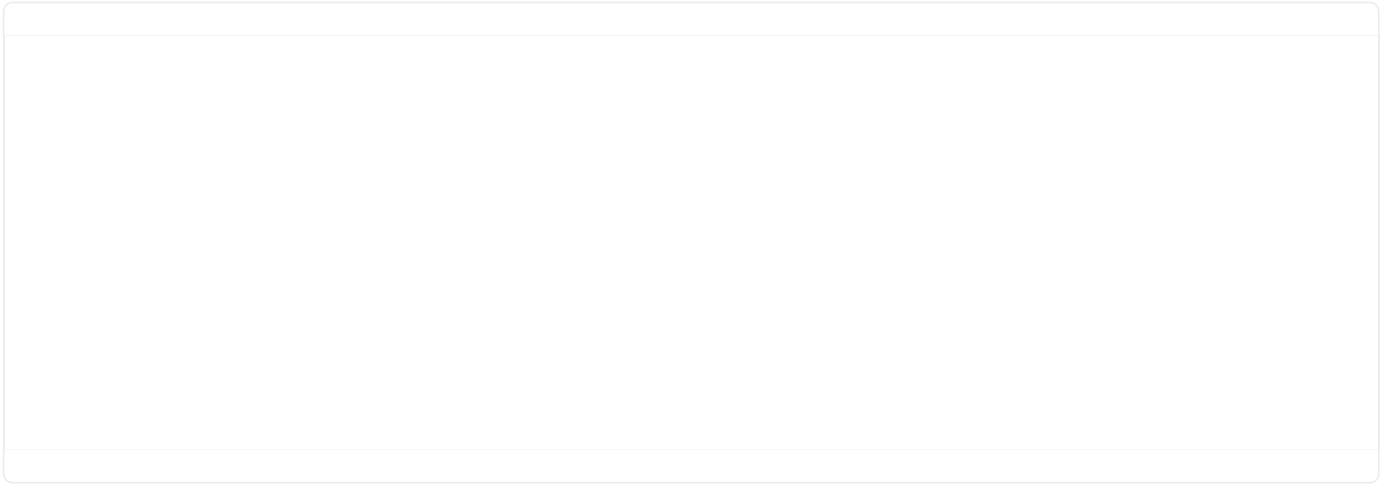
등록

글쓰기

답글

목록

▲ TOP



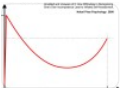
'| 일반 소스 공유 |' 게시판 글

전체 [미니게임] 말머리 글 이 게시판 새글 구독하기 ☐

[미니게임] 코인토스 🎮 [16]	EliF	2020.05.27.
[미니게임] 인디언 포커 Ver 1.1.1 [33]	BennyK	2020.05.27.
[미니게임] 미니게임 종합세트 (퀴즈, 베라, 369, 타자게임) 🎮 [13]	도미 doami2005	2020.05.27.
즉석복권 🎮 [12]	EliF	2020.05.27.
인디언 포커 (로 추정되는 것) [7]	BennyK	2020.05.27.

1 2 3 전체보기

이 카페 인기글

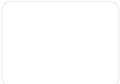


카톡봇 사람들은 꼭봐야됨

로코가 뭐길래 사람들이 환장하는걸까?

클칠

1 18



메신저R카톡방 들어와 채팅치면 환영해주

기기 성능

은당이

0 2



웹툰

카봇커 채팅방을 그룹으로 만들면 오실분?

성빈

1 52

원주율 100자리 가져오기

T봇

0 6



간단 계산기 입니다

deadmau5

0 7

ArtistBRIN

0 7

1 2 3 4 5