

.매장 공간 검색을 시 권리 연결을 연결합니다.(on~on/off~on)

-/방목록

:등록된 방들과 전원 상태를 mainroom에 표시합니다.

-/유저설정 [방이름],[유저이름],[등급]

:해당 방과 유저가 존재하면 입력된 등급으로 유저의 등급을 지정합니다.

-/전체저장

:방목록과 각 방의 공지를 백업파일에 저장해둡니다.

(v0.5에 유저파일 백업도 포함예정)

2. 등록된 방

<admin 등급>

-/부어드민등록 [유저이름]

:user 등급의 해당 유저가 존재할 시 부어드민(staff)의 등급을 부여합니다.

-/부어드민해제 [유저이름]

:staff 등급의 해당 유저가 존재할 시 user의 등급을 부여합니다.

-/정지

:/공지,/프카 의 사용을 정지합니다.

-/재개

:/공지,/프카 의 사용을 허용합니다.

-/공지설정 [공지내용]

:/공지 의 내용을 설정합니다.

-/전원

:해당 방 봇의 전원을 on/off 합니다.

(off일 동안은 /전원 외 명령어에 반응안하며 유저 등록도 비활성화, 리로드 및 전체저장은 활성화)

<staff 이상 등급>

-/공지

:공지에 저장된 내용을 표시합니다.

-/프카

:오픈톡방용으로 프로필카드 설정 방법을 안내합니다.

-/저장

:해당 방 유저 정보를 유저 파일에 저장합니다.

-/도움말

:사용가능한 명령어 목록을 표시합니다.

-/정보

:봇의 버전과 등록된 방 이름, 유저목록과 공지를 표시합니다.

- /리로드 나 /전체종료 시 작업 수행 이전 모든 정보를 저장,백업하기
- 일정시간 이상 유저 활동이 없을 시 유저목록에서 해당 유저 제거하기
- mainroom에서 직접 등록된 방의 유저 제거 기능 추가하기
- 휴대폰 배터리 잔량과 충전상태에 따라 자동으로 정보 저장,알림 후 종료하기

정도로 생각하고 있으며 이런저런 오류 발견 시 해당 사항 수정 및 구조의 체계화도 해야겠습니다..

각 등급에 해당하는 명령어들은 소스 서두에 있는 ctrlCommands(관리방 명령어), adminCommands(admin 명령어), maintainaceCommands(staff 이상 명령어)에 추가할 수 있으며 해당 명령어에 대한 작동은 //관리방 루틴, 각 //어드민 이상 권한 과 //스태프 이상 권한으로 분리된 각 구역의 switch의 case로 자유롭게 추가하실 수 있습니다.
(등급도 추가하실 수는 있습니다. 권한 부분은 크게 복사붙혀넣기 후 수정이 거쳐지긴 하겠지만..)

단 명령어 내에 띄어쓰기가 존재하면 안되며, 명령어에 추가적으로 기입되는 인자는 명령어와 띄어쓰기가 되어야합니다.

ex)

올바른 예: /<명령어> [인자]

나쁜 예: /<명> <령어> [인자], /<명령어>[인자]

<해당 소스는 인자를 제외한 명령어에만 반응하니, 추가적인 정보에 대한 처리는 직접 소스를 짜셔야 합니다.>

- 현재로서는 유저의 이름만을 식별 기준으로 삼기에 보안상의 문제가 있을 수 있습니다.
- 야매코딩만 해온 야매입니다. 원갈 해야겠다고 생각해내고(아주 중요합니다. 생각을 했어요) 만든 첫 소스로, 이해하기 상당히 난해하며 불필요한 부분이나 간소화 될 수 있는 부분이 상당 수 존재할 수 있습니다.(간소화가 가능한 파트와 방법을 남겨주신다면 제 사랑 한움큼 쥐어드릴게요)
- 간간히 시간 날때마다 이것저것 수정하고 추가하고 제거하며 업데이트 일지나 올려겠습니다.

흑흑 지나가신 분들도 있겠지만 그럼에도 이 잉여로운 글의 첫 문단의 첫 문장이라도 읽어주셨음에 감사드립니다.

아래는 소스입니다. 난잡하게 만들어져 상당히 길고 난장판입니다. (하하하-먼산)

```

1  const scriptName = "Rootbot v0.4";
2
3  const mainroom = "";
4
5  var d = new Date();
6
7  var rooms = ["null"];
8  var users = [[["null","null",0]]];
9  var power = [false],started = [true],emerstop = [false],utemp = [],ntemp = [0], n = [0];
10 var usetup = "",temp = " ";
11 var roomindex = 0;
12 var chunk = 3;
13
14 var a =[];
15
16 var levCheck = "",userindex = 0, tuserindex=-1, userexist = 0;
17
18 let full = "\u200b".repeat(500);
19 let normal = '\u25A3[루봇]';
20 let warn = '[경고]';
21 let system = '[시스템]';
22 let sep = "|---|";
23
24 var throwOnError:

```

```

25 var temp1,temp2,temp3,temp4,temp5,temp6,i,j,k,l,powerstat,indicator;
26 var boot = true;
27
28 var positions = ["user","staff","admin"];
29 var ctrlCommands = ["/리로드","/전체종료","/방추가","/방제거","/방정보","/방목록","/유저설정","/빙
30 var adminCommands = ["/부어드민등록","/부어드민해제","/정지","/재개","/공지설정","/전원"];
31 var maintananceCommands = ["/공지","/프카","/자장","/도움말","/정보"];
32
33 var notification = [];
34
35 var OUT = {
36   notify : function() { Api.replyRoom(rooms[roomindex],normal+':'+ 공지입니다.\n'+full+notifi
37   profile : function() { Api.replyRoom(rooms[roomindex],normal+':'+ 프카 설정법입니다.\n'+full
38   help : function() { Api.replyRoom(rooms[roomindex],normal+': '+{명령어}'+full+'\n\n'+<관리
39
40 };
41
42 function startManager(directory) {
43   if(!DataBase.getDataBase(directory)) {
44     DataBase.setDataBase(directory,"");
45     Api.replyRoom(mainroom, normal+': '+directory+' 생성 완료.');
```

```

93     emerstop.splice(i,0,true);
94     power.splice(i,0,false);
95     ntemp.splice(i,0,0);
96     n.splice(i,0,0);
97     started.splice(i,0,true);
98     if(DataBase.getDataBase("DB/users/"+rooms[i])) {
99         var tempa = DataBase.getDataBase("DB/users/"+rooms[i]);
100         var tempaa = tempa.split(sep);
101         // Api.replyRoom(mainroom,started[i]+'/');
102         users[i]=["null","null",0];
103         for(j=chunk,k=tempaa.length;j<k;j+=chunk) {
104             users[i].splice(j/chunk,0,tempaa.slice(j,j+chunk));
105         }
106         usetup = "유저 파일 지정 완료.";
107     }
108     else {
109         DataBase.setDataBase("DB/users/"+rooms[i],"");
110         usetup = "유저 파일 생성 완료.";
111     }
112
113     Api.replyRoom(rooms[i], normal+': \n<'+rooms[i]+'> :'+usetup);
114 }
115 // Api.replyRoom(mainroom,started+'/'+power+'/'+users);
116 }
117
118
119 function response(room, msg, sender, isGroupChat, replier, imageDB) {
120
121     var Sender = sender;
122
123     if(boot) {
124         rooms=startManager("DB/rooms");
125         notification=startManager("DB/notification");
126         boot=!boot;
127         for(i=0,l=rooms.length;i<l;i++) {
128             started[i]=true;
129             power[i]=false;
130             emerstop[i]=false;
131         }
132         for(i=1,j=rooms.length;i<j;i++) {
133             Api.replyRoom(rooms[i],normal+': '+system+': '+scriptName+' 온라인');
134         }
135     }
136 }
137
138 if(rooms.indexOf(room)!=-1) {
139     roomindex=rooms.indexOf(room);
140 }
141
142
143 //초기설정
144 for(i=0,l=rooms.length;i<l;i++) {
145     if (started[i]==true) {
146         if(DataBase.getDataBase("DB/users/"+rooms[i])) {
147             var tempa = DataBase.getDataBase("DB/users/"+rooms[i]);
148             // Api.replyRoom(mainroom,tempa);
149             var tempaa = tempa.split(sep);
150             // Api.replyRoom(mainroom,tempaa);
151             users[i] = ["null","null",0];
152             for(j=chunk,k=tempaa.length;j<k;j+=chunk) {
153                 users[i].splice(j/chunk,0,tempaa.slice(j,j+chunk));
154             }
155             usetup = "유저 파일 불러오기 완료";
156         }
157         else {
158             usetup = "빈 유저 파일";
159         }
160         Api.replyRoom(mainroom, normal+': \n<'+rooms[i]+'> :'+usetup);
161         started[i]=false;
162     }

```

```

162     }
163 }
164 // Api.replyRoom(mainroom,users);
165
166 //관리방 루틴
167 if(room==mainroom) {
168
169     var indicator = msg.split(" ")[0];
170
171     if(ctrlCommands.indexOf(indicator)!=-1) {
172
173         switch(ctrlCommands.indexOf(indicator)) {
174
175             //리로드
176             case 0 :
177                 Api.replyRoom(mainroom,normal+': 리로드 시작');
178                 for(i=1,j=rooms.length;i<j;i++) {
179                     Api.replyRoom(rooms[i],normal+':'+system+':'+warn+' : 리로드');
180                 }
181                 Api.reload(scriptName, throwOnError=false);
182                 break;
183
184             //전체종료
185             case 1 :
186                 Api.replyRoom(mainroom,normal+':'+system+' : 종료합니다');
187                 for(i=1,j=rooms.length;i<j;i++) {
188                     Api.replyRoom(rooms[i],normal+':'+system+' : 종료합니다');
189                 }
190                 Api.off(scriptName);
191                 break;
192
193             //방추가
194             case 2 :
195                 temp4=msg.slice(4).trim();
196                 if(rooms.indexOf(temp4)!=-1) {
197                     Api.replyRoom(mainroom, normal+':'+warn+' : '+'해당 방<'+temp4+'>은 이미 등록되
198                 }
199                 else {
200                     rooms.push(temp4);
201                     notification.push("");
202                     emerstop.push(false);
203                     ntemp.push(0);
204                     n.push(0);
205                     users.push(["null","null",0]);
206                     started.push(true);
207                     power.push(false);
208                     Api.replyRoom(mainroom, normal+': '+'<'+temp4+'> 등록이 성공적으로 완료되었습니다
209                     roomDBUpdate();
210                     Api.replyRoom(mainroom,rooms.indexOf(temp4)+'/'+users[rooms.indexOf(temp4)]+'
211                     locDBUpdate(rooms.indexOf(temp4));
212                 }
213                 break;
214
215             //방제거
216             case 3 :
217                 temp4=msg.slice(4).trim();
218                 if(rooms.indexOf(temp4)!=-1) {
219                     var l = rooms.indexOf(temp4);
220                     notification.splice(l,1);
221                     emerstop.splice(l,1);
222                     rooms.splice(l,1);
223                     ntemp.splice(l,1);
224                     n.splice(l,1);
225                     users.splice(l,1);
226                     started.splice(l,1);
227                     power.splice(l,1);
228                     Api.replyRoom(mainroom, normal+': '+'<'+temp4+'> 삭제가 성공적으로 완료되었습니다
229                     roomDBUpdate();
230                 }

```

```

231     else {
232         Api.replyRoom(mainroom, normal+':'+warn+': '+'해당 방<'+temp4+'>은 존재하지 않
233     }
234     break;
235
236 //방정보
237 case 4 :
238     temp4=msg.slice(4).trim();
239     if(rooms.indexOf(temp4)!=-1) {
240         var k=rooms.indexOf(temp4);
241         temp6 = [];
242         // Api.replyRoom(mainroom,users);
243         for(i=0,j=users[k].length;i<j;i++) {
244             temp6.push(users[k][i].join(" | "));
245         }
246         Api.replyRoom(mainroom, normal+': '+'\n<'+temp4+'>'+'\n유저 목록:\n'+full+tem
247         Api.replyRoom(mainroom, normal+': '+'\n<'+temp4+'>'+'\n공지:\n'+full+notifica
248     }
249     else {
250         Api.replyRoom(mainroom, normal+':'+warn+': '+'해당 방<'+temp4+'>은 존재하지 않
251     }
252     break;
253
254 //방목록
255 case 5 :
256     if(rooms.length<2) {
257         Api.replyRoom(mainroom, normal+':'+warn+': '+'존재하는 방이 없습니다.');
```

```

300         break;
301
302     //방 전원
303     case 7 :
304         temp4=msg.slice(4).trim();
305         if(rooms.indexOf(temp4)!=-1) {
306             k=rooms.indexOf(temp4);
307             if(power[k]) {
308                 power[k]=false;
309                 powerstat = "꺼";
310             }
311             else {
312                 power[k]=true;
313                 powerstat = "켜";
314             }
315             Api.replyRoom(mainroom, normal+': '+'\n<'+temp4+'> 이(가) '+powerstat+'졌습니다
316             Api.replyRoom(rooms[k], normal+':'+system+': '+'\n<'+temp4+'> 이(가) '+powerstat+
317             }
318             else {
319                 Api.replyRoom(mainroom, normal+':'+warn+': '+'해당 방<'+temp4+'>은 존재하지 않
320             }
321             break;
322
323     //전체 저장
324     case 8 :
325         Backup();
326         break;
327
328     default :
329         break;
330
331
332     }
333 }
334 }
335 // Api.replyRoom(mainroom,users[roomindex]);
336 //단체방 반응 루틴
337 if(rooms.indexOf(room)!=-1) {
338     if(power[roomindex]==true) {
339         n[roomindex] = d.getHours();
340
341
342
343
344     //자동유저추가
345     userexist=0;
346     // Api.replyRoom(mainroom,users[roomindex]);
347     for(j=0,k=users[roomindex].length;(j<k)&&(userexist==0);j++) {
348         if(users[roomindex][j].indexOf(Sender)!=-1) {
349             userexist=1;
350         }
351     }
352     if (userexist==0) {
353         users[roomindex].push([Sender,"user",0]);
354         Api.replyRoom(rooms[roomindex],normal+': '+'Sender+\'님, 환영합니다!');
355         locDBUpdate(roomindex);
356     }
357
358     //자동 백업
359     if(n[roomindex]!=ntemp[roomindex]) {
360         locDBBackup(roomindex);
361         ntemp[roomindex]=n[roomindex];
362     }
363
364     //유저 권한 확인
365     for(j=0,k=users[roomindex].length;j<k;j++) {
366         if(users[roomindex][j].indexOf(Sender)!=-1) userindex=j;
367     }
368     levCheck=users[roomindex][userindex][1];

```



```

369
370
371 //어드민 이상 권한
372 if (levCheck=="admin") {
373
374     indicator = msg.split(" ")[0];
375
376     if(adminCommands.indexOf(indicator)!=-1) {
377
378         switch(adminCommands.indexOf(indicator)) {
379
380             //부어드민 등록
381             case 0 :
382                 var temp1 = msg.slice(8).trim();
383                 tuserindex=-1;
384                 for(j=0,k=users[roomindex].length;j<k&&tuserindex==-1;j++) {
385                     if(users[roomindex][j].indexOf(temp1)!=-1) tuserindex=j;
386                 }
387                 if(tuserindex==-1){
388                     Api.replyRoom(rooms[roomindex],normal+':'+warn+': '+temp1+'님은 존재하지 않는
389                 }
390                 else if(users[roomindex][tuserindex][1]==positions[1]) {
391                     Api.replyRoom(rooms[roomindex],normal+':'+warn+': '+temp1+'님은 이미 부어드민
392                 }
393                 else if(users[roomindex][tuserindex][1]==positions[2]) {
394                     Api.replyRoom(rooms[roomindex],normal+':'+warn+': '+temp1+'님은 관리자입니다.
395                 }
396                 else if(users[roomindex][tuserindex][1]==positions[0]){
397                     Api.replyRoom(rooms[roomindex],normal+': '+temp1+'님이 부어드민으로 등록되었습
398                     users[roomindex][tuserindex][1]=positions[1];
399                     locDBUpdate(roomindex);
400                 }
401                 else {
402                     Api.replyRoom(mainroom,normal+':'+warn+': \n<'+rooms[roomindex]+'>\n'+temp1
403                 }
404                 break;
405
406             //부어드민 해제
407             case 1 :
408                 var temp2 = msg.slice(8).trim();
409                 tuserindex=-1;
410                 for(j=0,k=users[roomindex].length;j<k&&tuserindex==-1;j++) {
411                     if(users[roomindex][j].indexOf(temp2)!=-1) tuserindex=j;
412                 }
413                 if(tuserindex==-1){
414                     Api.replyRoom(rooms[roomindex],normal+':'+warn+': '+temp2+'님은 존재하지 않는
415                 }
416                 else if(users[roomindex][tuserindex][1]==positions[0]) {
417                     Api.replyRoom(rooms[roomindex],normal+':'+warn+': '+temp2+'님은 부어드민이 이
418                 }
419                 else if(users[roomindex][tuserindex][1]==positions[2]) {
420                     Api.replyRoom(rooms[roomindex],normal+':'+warn+': '+temp2+'님은 관리자입니다.
421                 }
422                 else if(users[roomindex][tuserindex][1]==positions[1]){
423                     Api.replyRoom(rooms[roomindex],normal+': '+temp2+'님이 부어드민에서 해제되었습
424                     users[roomindex][tuserindex][1]=positions[0];
425                     locDBUpdate(roomindex);
426                 }
427                 else {
428                     Api.replyRoom(rooms[roomindex],normal+':'+warn+': '+temp2+'님의 권한 데이터어
429                     Api.replyRoom(mainroom,normal+':'+warn+': '+rooms[roomindex]+'+':'+temp2+'님의
430                 }
431                 break;
432
433             //알림 정지
434             case 2 :
435                 emerstop[roomindex]=true;
436                 Api.replyRoom(rooms[roomindex],normal+': '+알림을 정지하였습니다. ');
437                 break;

```

```

438
439 //알림 재개
440 case 3 :
441     emerstop[roomindex]=false;
442     Api.replyRoom(rooms[roomindex],normal+': '+'알림을 재개하였습니다. ');
443     break;
444
445 //공지 설정
446 case 4 :
447     var temp3 = msg.slice(5).trim();
448     if(temp3!="") {
449         notification[roomindex]=temp3;
450         notifiDBUpdate();
451     }
452     else {
453         notification[roomindex]=normal+': '+warn+': '+'설정된 공지가 없습니다. \' /공지
454     }
455     Api.replyRoom(rooms[roomindex],normal+': '+'공지가 설정되었습니다. ');
456     break;
457
458 //전원
459 case 5 :
460     power[roomindex]=false;
461     Api.replyRoom(rooms[roomindex], normal+': '+'\n전원이 꺼졌습니다. ');
462     Api.replyRoom(mainroom, normal+': '+system+': '+'\n<'+rooms[roomindex]+'> 이(가
463     break;
464
465 default:
466     break;
467 }
468 }
469 }
470
471 //스태프 이상 권한
472 if (levCheck=="admin" || levCheck=="staff") {
473
474     indicator = msg.split(" ")[0];
475
476     if(maintananceCommands.indexOf(indicator)!=-1) {
477
478         switch (maintananceCommands.indexOf(indicator)) {
479
480             //공지
481             case 0 :
482                 if(!emerstop[roomindex]) {
483                     OUT.notify();
484                 }
485                 break;
486
487             //프카
488             case 1 :
489                 if(!emerstop[roomindex]) {
490                     OUT.profile();
491                 }
492                 break;
493
494             //저장
495             case 2 :
496                 locDBBackup(roomindex);
497                 break;
498
499             //도움말
500             case 3 :
501                 OUT.help();
502                 break;
503
504             //정보
505             case 4 :
506                 temp6 = [];

```

```

500         temp6 = [];
507         for(i=0,j=users[roomindex].length;i<j;i++) {
508             temp6.push(users[roomindex][i].join(" | "));
509         }
510         Api.replyRoom(rooms[roomindex], '[봇]:'+scriptName);
511         Api.replyRoom(rooms[roomindex], '<'+rooms[roomindex]+'>');
512         Api.replyRoom(rooms[roomindex], '유저 목록:\n'+full+temp6.join("\n"));
513         Api.replyRoom(rooms[roomindex], '공지:\n'+full+notification[roomindex]);
514         break;
515
516         default :
517             break;
518     }
519 }
520 }
521 }
522 else {
523     userexist=0;
524     for(j=0,k=users[roomindex].length;(j<k)&&(userexist==0);j++) {
525         if(users[roomindex][j].indexOf(Sender)!=-1) {
526             userexist=1;
527         }
528     }
529     if(userexist==1) {
530         for(j=0,k=users[roomindex].length;j<k;j++) {
531             if(users[roomindex][j].indexOf(Sender)!=-1) userindex=j;
532         }
533         levCheck=users[roomindex][userindex][1];
534     }
535     if (levCheck=="admin") {
536         if(msg.indexOf("/전원")==0) {
537             power[roomindex]=true;
538             Api.replyRoom(rooms[roomindex], normal+' : '+'\n전원이 켜졌습니다. ');
539             Api.replyRoom(mainroom, normal+' : '+system+' : '+'\n<'+rooms[roomindex]+'> 이(가) 켜졌
540         }
541     }
542 }
543 }
544 }

```

추가1:

22번줄에 선언된 sep는 후에 const로 변경될것입니다만, 파일로 정보 저장 후 다시 읽어들일 시 정보 구분의 기준이 되는 seperator입니다. 그렇기에 임의로 방을 직접 추가하시거나 특정 방에 유저를 추가하신다면(이후에는 유저 추가/제거 기능도 넣을겁니다.) DB/rooms 혹은 DB/users/[방이름]으로 저장된 txt파일 내에서 sep를 고려하여 직접 수정이 가능합니다.

각 유저는 총 3개의 정보를 지니게 되는데, 이름, 등급, 0(후에 사용 예정) 의 3개의 정보가 한 단위로 이루어집니다. 파일 저장시에는 이를 일렬로 나열하여 사이사이 구분선만 넣어주고, 후에 읽어들일 시 3단위로 끊어서 읽기에 sep와 3단위 구조를 고려하지 않은 유저등록 혹은 파일 조작은 오류를 일으킬 수 있습니다.



Emera—d님의 게시글 더보기 >

❤️ 좋아요 7 💬 댓글 5

🔗 공유 | 신고

Violet
와 onStartCompile 저렇게 알뜰하게 쓰시는 분은 또 처음보네요 ㄷㄷ
2020.01.22. 01:16 답글쓰기

AlphaDo
저도 5줄 이상은 안써봤는덴..
2020.01.22. 06:54 답글쓰기

호이이ㄴㅇㅇ
사용해보니까 파일 생성만 하고 다른건 안되는데 어케상ㅇㅇ하나용
2020.01.22. 11:40 답글쓰기

링디
감사합니다 :)
2020.01.23. 14:09 답글쓰기

초초초초보컴맹
감사합니다~이건 어느때 사용하는거예요?
2020.06.03. 14:46 답글쓰기

Hibot
댓글을 남겨보세요



등록

글쓰기

답글

목록

▲ TOP

'| 일반 소스 공유 |' 게시판 글

이 게시판 새글 구독하기

cpu 코어 개수 가져오기 [7]	쿨칠	2020.01.22.
세이프 이발의 다음 버전, <EasyEval> 다운로드 [22]	OnTheWay	2020.01.22.
첫 카톡봇 소스- 룯봇 [5]	Emera—d	2020.01.22.
대 — 총 만든 우리말샘 검색기 [4]	구름	2020.01.21.
bing 날씨 정보 [15]	꾸와	2020.01.20.

이 카페 인기글

간편 자동응답과 한글코딩을 메인으로 제공
하는 카톡봇앱 개발 예정

성빈
♡1 💬16



링크를 포함한 글 홍보시 링크 분석(?)하기

안녕하세요!
♡0 💬2
트위터봇
♡0 💬6

요즘 나타나는 새로운 홍보 수법 중 하나가

Dark Tornado
♡0 💬7

타 메신저 사용시 무한반복

NaN
♡0 💬1



"시초가 중국발" 발언 관련 해명글 및 부탁

반가워요.

천방지축하연
♡0 💬4



태양, 달 정보 구현 완료

Dark Tornado
♡0 💬10
오른아발 괴롭히기(메봇R)
ERROR
♡0 💬7