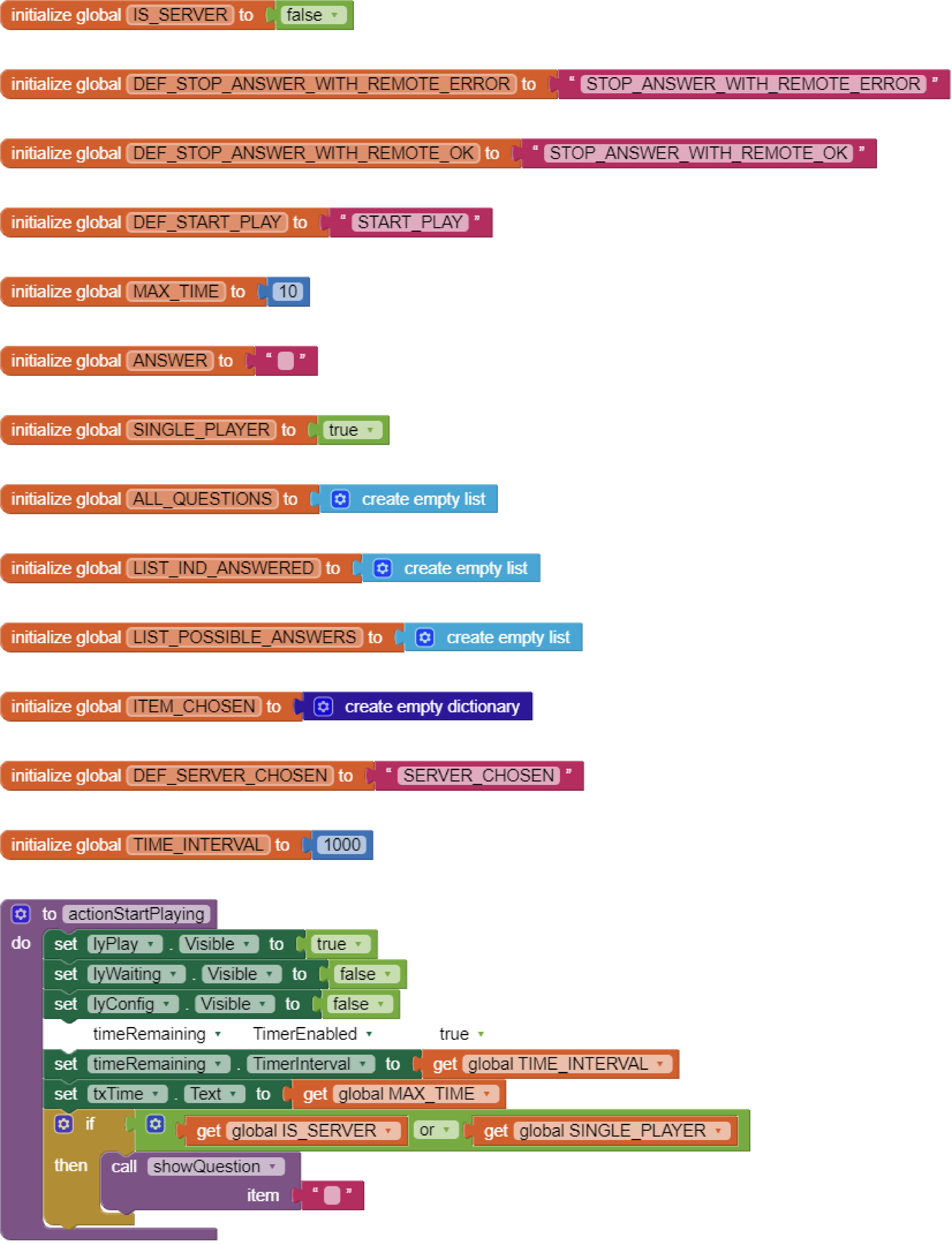
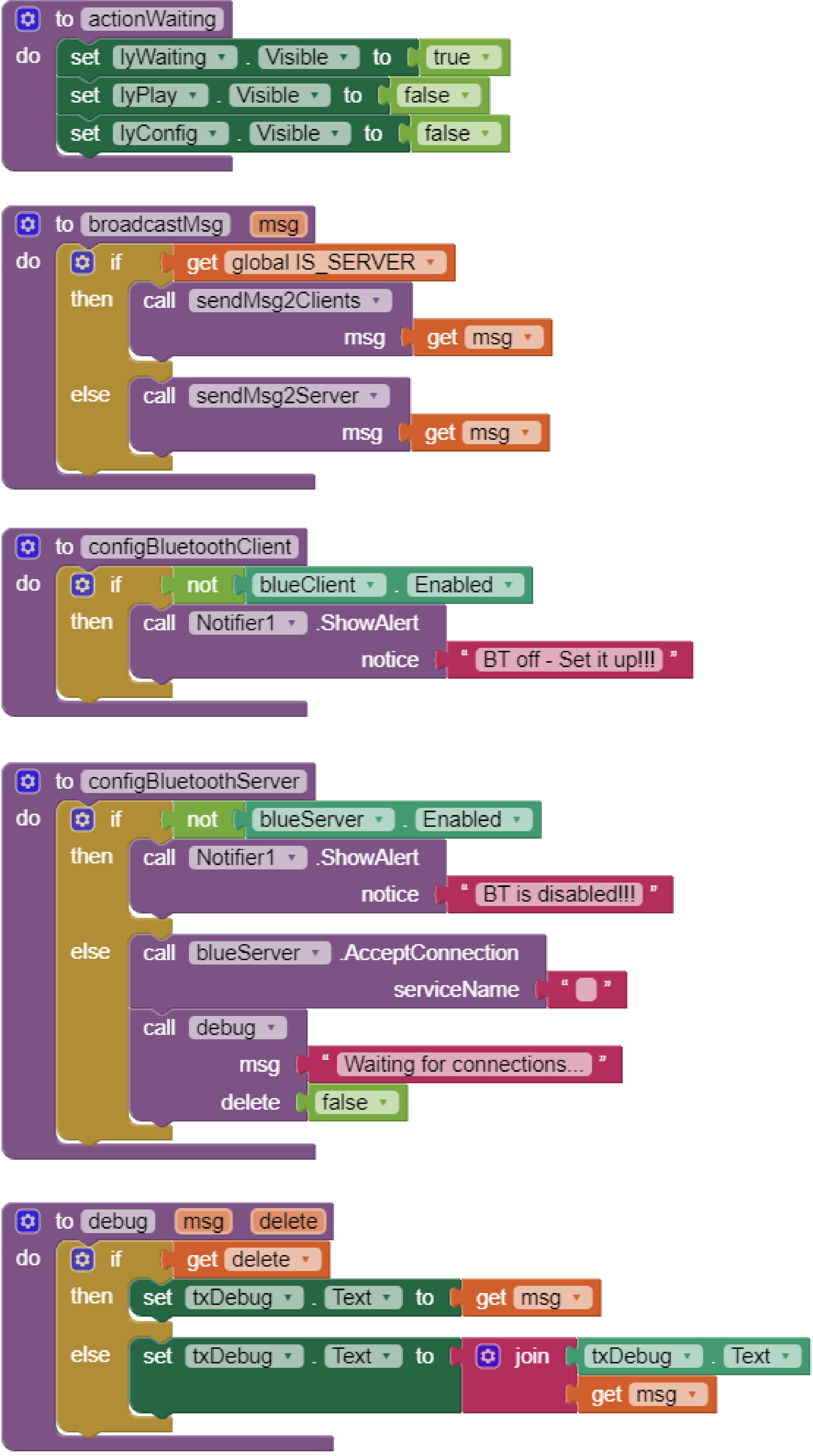
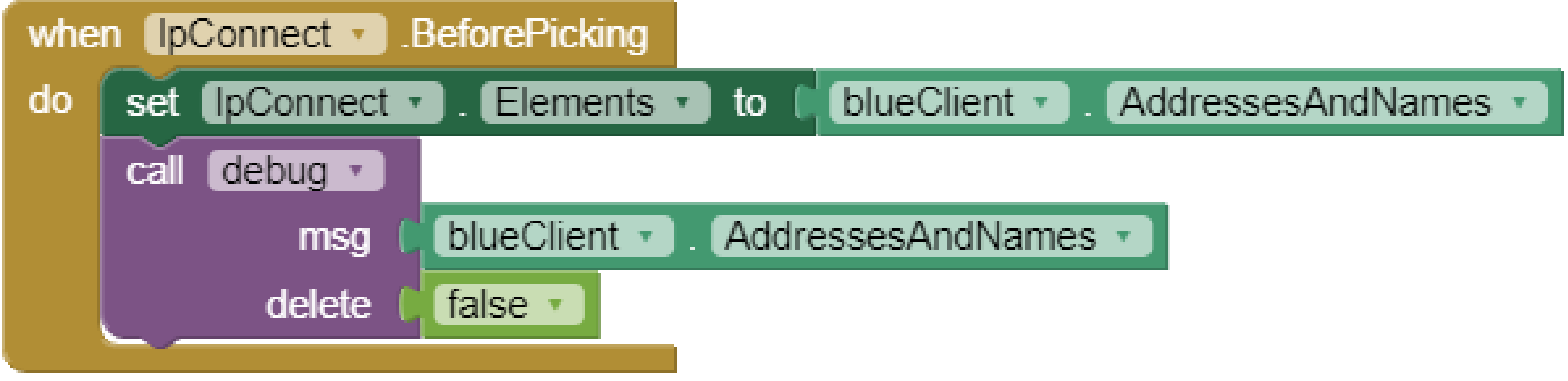


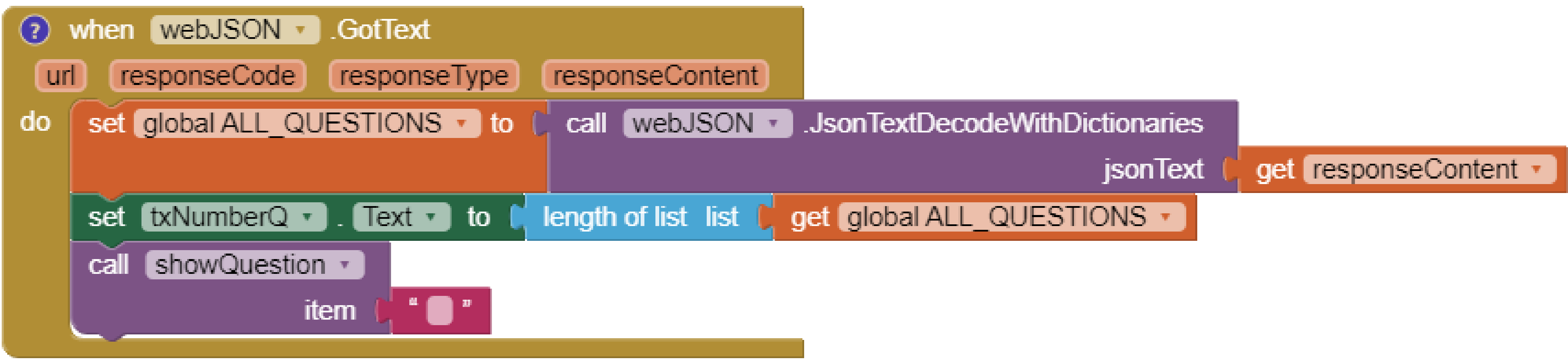
```
initialize global IS_SERVER to false

initialize global DEF_STOP_ANSWER_WITH_REMOTE_ERROR to " STOP_ANSWER_WITH_REMOTE_ERROR "

initialize global DEF_STOP_ANSWER_WITH_REMOTE_OK to " STOP_ANSWER_WITH_REMOTE_OK "

initialize global DEF_START_PLAY to " START_PLAY "

initialize global MAX_TIME to 10

initialize global ANSWER to " "

initialize global SINGLE_PLAYER to true

initialize global ALL_QUESTIONS to create empty list

initialize global LIST_IND_ANSWERED to create empty list

initialize global LIST_POSSIBLE_ANSWERS to create empty list

initialize global ITEM_CHOSEN to create empty dictionary

initialize global DEF_SERVER_CHOSEN to " SERVER_CHOSEN "

initialize global TIME_INTERVAL to 1000

to actionStartPlaying
do  set lyPlay . Visible to true
    set lyWaiting . Visible to false
    set lyConfig . Visible to false
        timeRemaining . TimerEnabled    true
    set timeRemaining . TimerInterval to get global TIME_INTERVAL
    set txTime . Text to get global MAX_TIME
    if   get global IS_SERVER or get global SINGLE_PLAYER
    then call showQuestion
                      item " "
```



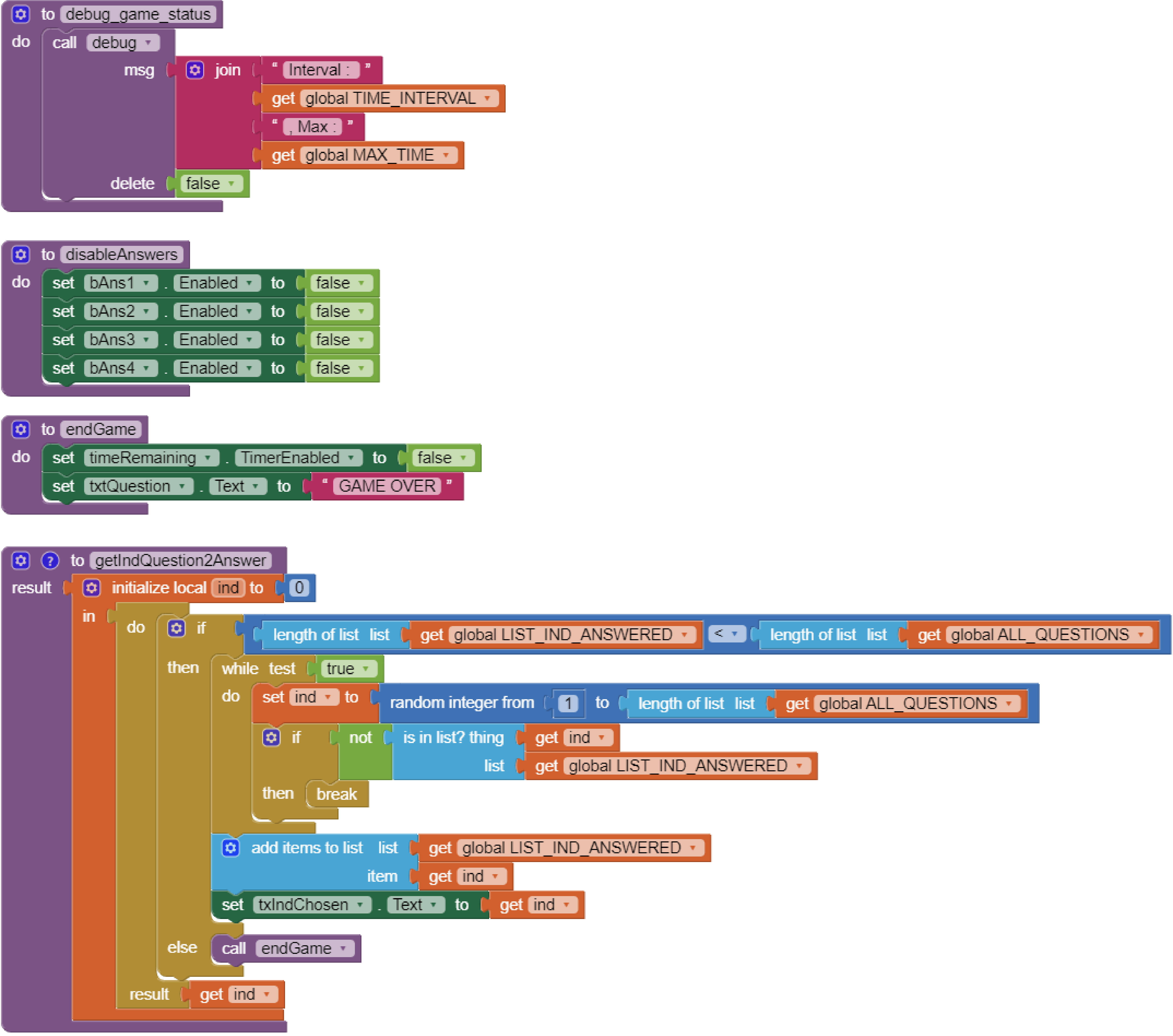
```
to actionWaiting
do  set lyWaiting . Visible to true
    set lyPlay . Visible to false
    set lyConfig . Visible to false

to broadcastMsg msg
do  if get global IS_SERVER
    then call sendMsg2Clients
              msg get msg
    else call sendMsg2Server
              msg get msg

to configBluetoothClient
do  if not blueClient . Enabled
    then call Notifier1 .ShowAlert
              notice " BT off - Set it up!!! "

to configBluetoothServer
do  if not blueServer . Enabled
    then call Notifier1 .ShowAlert
              notice " BT is disabled!!! "
    else call blueServer .AcceptConnection
              serviceName " "
         call debug
              msg " Waiting for connections... "
              delete false

to debug msg delete
do  if get delete
    then set txDebug . Text to get msg
    else set txDebug . Text to join txDebug . Text
                                    get msg
```



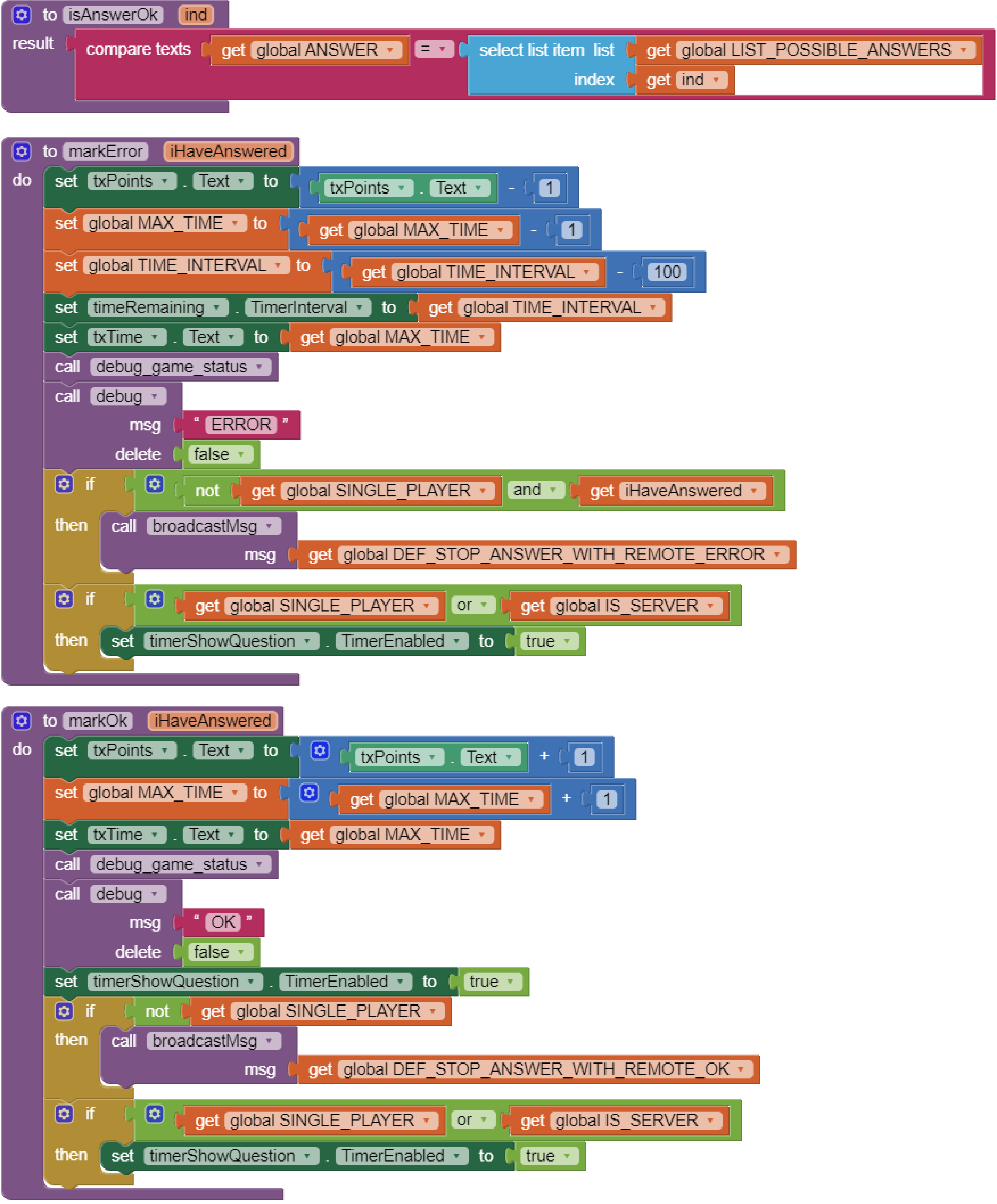
```
when IpConnect .BeforePicking
do  set IpConnect . Elements to  blueClient . AddressesAndNames
    call debug
               msg  blueClient . AddressesAndNames
            delete  false
```



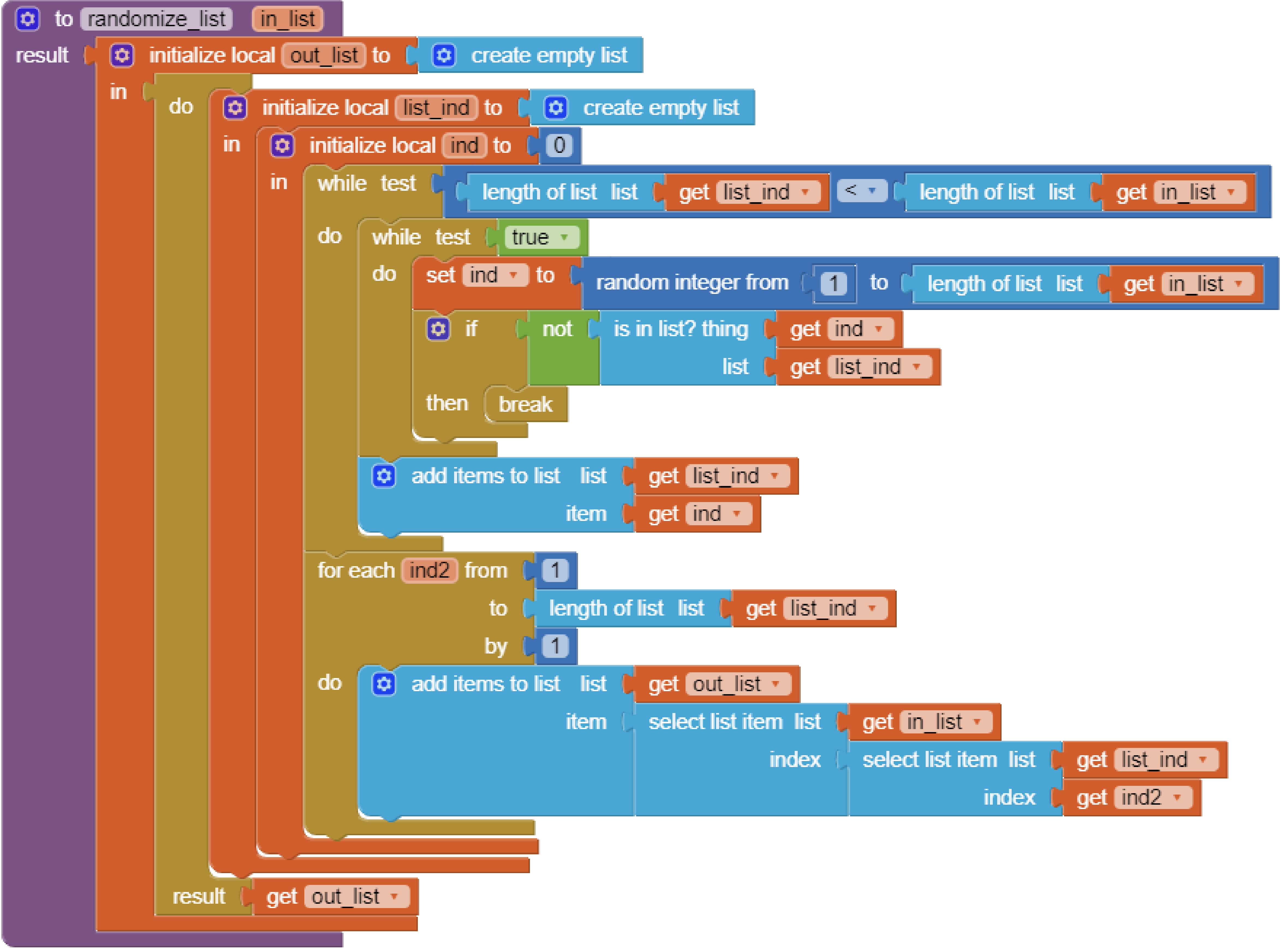
```
when webJSON .GotText
 url  responseCode  responseType  responseContent
do  set global ALL_QUESTIONS to  call webJSON .JsonTextDecodeWithDictionaries
                                                               jsonText  get responseContent
    set txNumberQ . Text to  length of list  list  get global ALL_QUESTIONS
    call showQuestion
                     item  " "
```



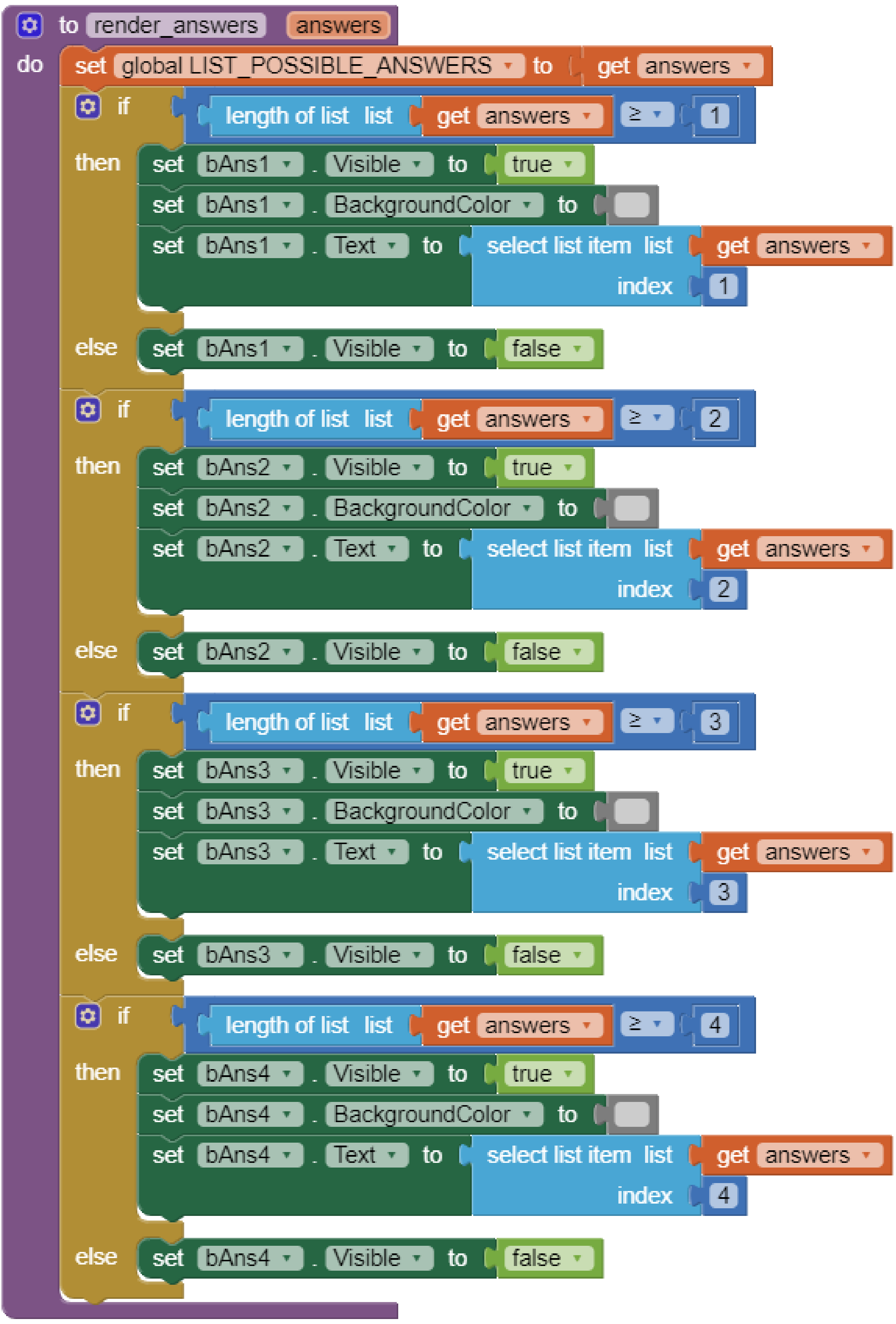
```
to debug_game_status
do  call debug
         msg  join  " Interval : "
                    get global TIME_INTERVAL
                    " , Max : "
                    get global MAX_TIME
         delete  false

to disableAnswers
do  set bAns1 . Enabled to  false
    set bAns2 . Enabled to  false
    set bAns3 . Enabled to  false
    set bAns4 . Enabled to  false

to endGame
do  set timeRemaining . TimerEnabled to  false
    set txtQuestion . Text to  " GAME OVER "

to getIndQuestion2Answer
result  initialize local ind to  0
        in  do  if   length of list  list  get global LIST_IND_ANSWERED  <  length of list  list  get global ALL_QUESTIONS
                then  while test  true
                      do  set ind to  random integer from  1  to  length of list  list  get global ALL_QUESTIONS
                          if  not  is in list? thing  get ind
                                                  list  get global LIST_IND_ANSWERED
                          then  break
                      add items to list  list  get global LIST_IND_ANSWERED
                                         item  get ind
                      set txIndChosen . Text to  get ind
                else  call endGame
            result  get ind
```



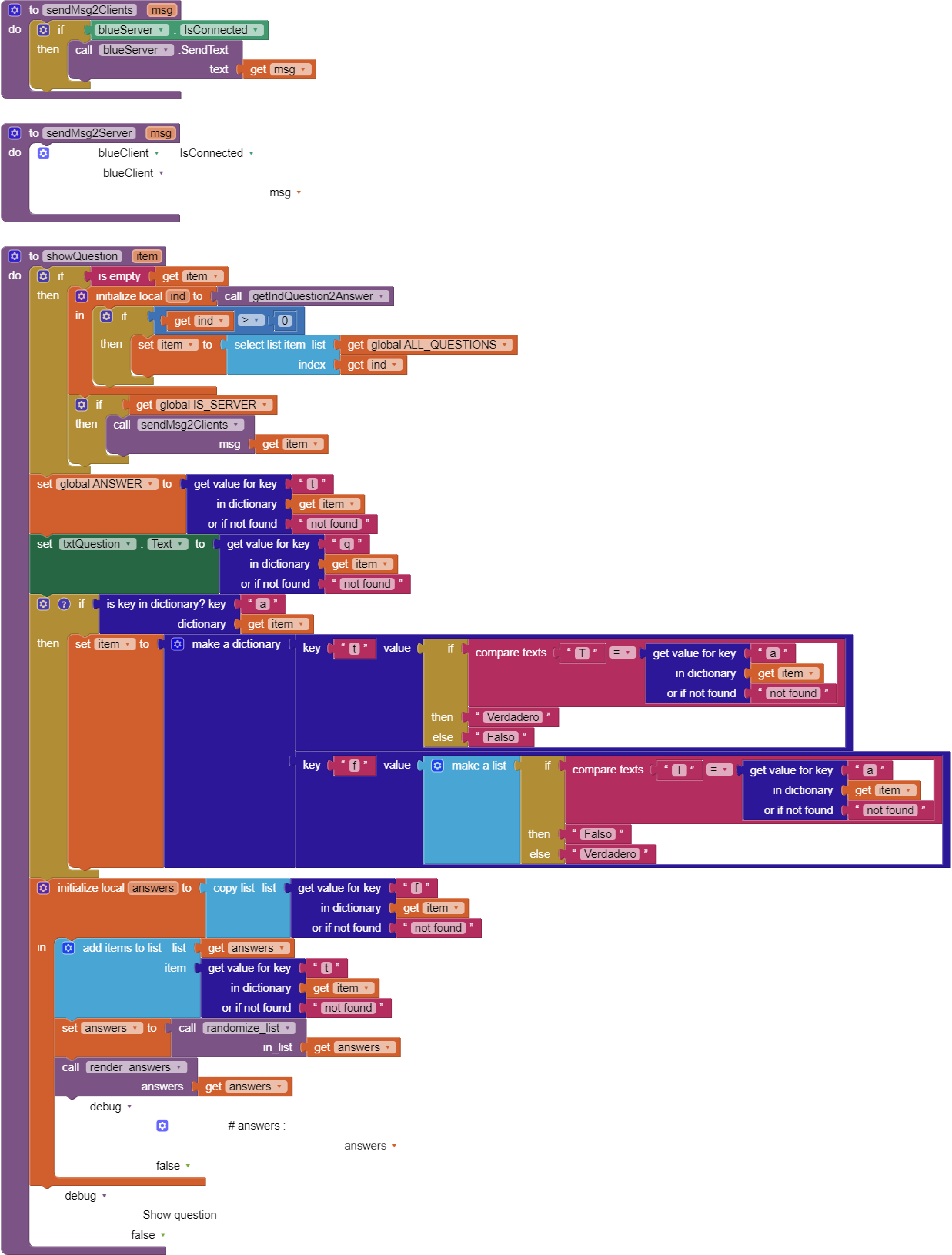
```
to isAnswerOk ind
result  compare texts  get global ANSWER  =  select list item  list  get global LIST_POSSIBLE_ANSWERS
                                                               index  get ind

to markError iHaveAnswered
do  set txPoints . Text to  txPoints . Text  -  1
    set global MAX_TIME to  get global MAX_TIME  -  1
    set global TIME_INTERVAL to  get global TIME_INTERVAL  -  100
    set timeRemaining . TimerInterval to  get global TIME_INTERVAL
    set txTime . Text to  get global MAX_TIME
    call debug_game_status
    call debug
            msg  " ERROR "
         delete  false
    if    not  get global SINGLE_PLAYER  and  get iHaveAnswered
    then  call broadcastMsg
                      msg  get global DEF_STOP_ANSWER_WITH_REMOTE_ERROR
    if    get global SINGLE_PLAYER  or  get global IS_SERVER
    then  set timerShowQuestion . TimerEnabled to  true

to markOk iHaveAnswered
do  set txPoints . Text to  txPoints . Text  +  1
    set global MAX_TIME to  get global MAX_TIME  +  1
    set txTime . Text to  get global MAX_TIME
    call debug_game_status
    call debug
            msg  " OK "
         delete  false
    set timerShowQuestion . TimerEnabled to  true
    if    not  get global SINGLE_PLAYER
    then  call broadcastMsg
                      msg  get global DEF_STOP_ANSWER_WITH_REMOTE_OK
    if    get global SINGLE_PLAYER  or  get global IS_SERVER
    then  set timerShowQuestion . TimerEnabled to  true
```



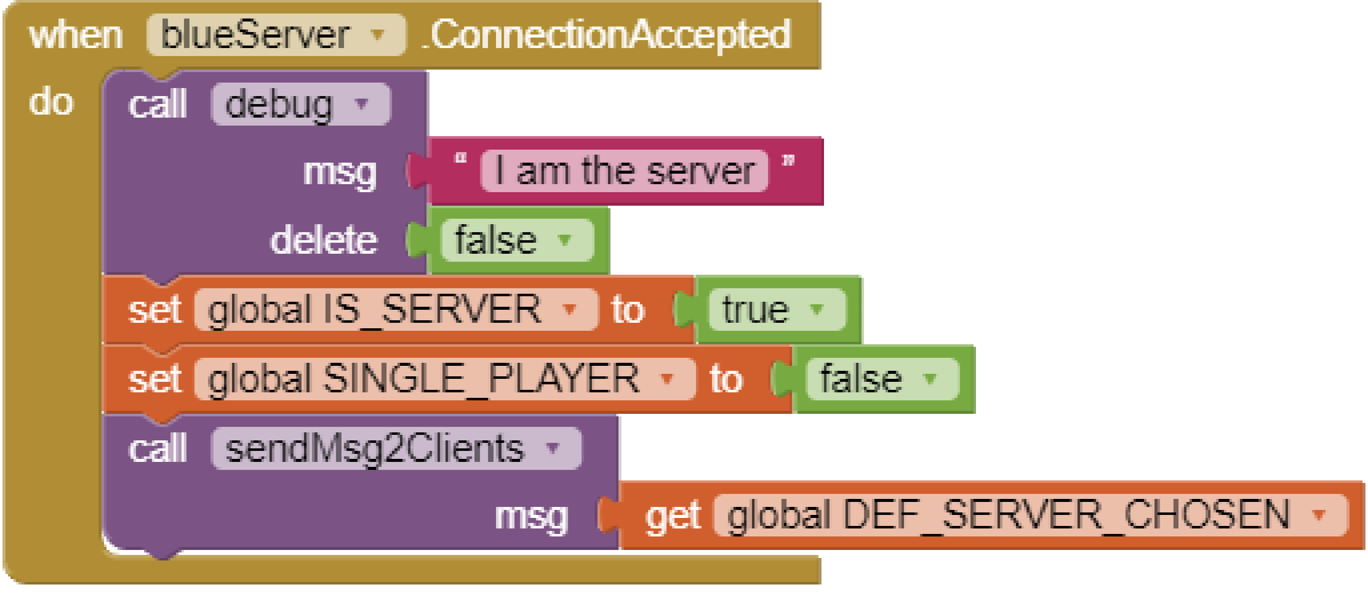
```
to randomize_list in_list
result  initialize local out_list to  create empty list
        in  do  initialize local list_ind to  create empty list
                in  initialize local ind to 0
                    in  while test  length of list list  get list_ind  <  length of list list  get in_list
                        do  while test  true
                            do  set ind to  random integer from 1 to  length of list list  get in_list
                                if  not  is in list? thing  get ind
                                                    list  get list_ind
                                then  break
                            add items to list list  get list_ind
                                              item  get ind
                        for each ind2 from 1
                                      to  length of list list  get list_ind
                                      by 1
                        do  add items to list list  get out_list
                                              item  select list item list  get in_list
                                                                   index  select list item list  get list_ind
                                                                                         index  get ind2
            result  get out_list
```



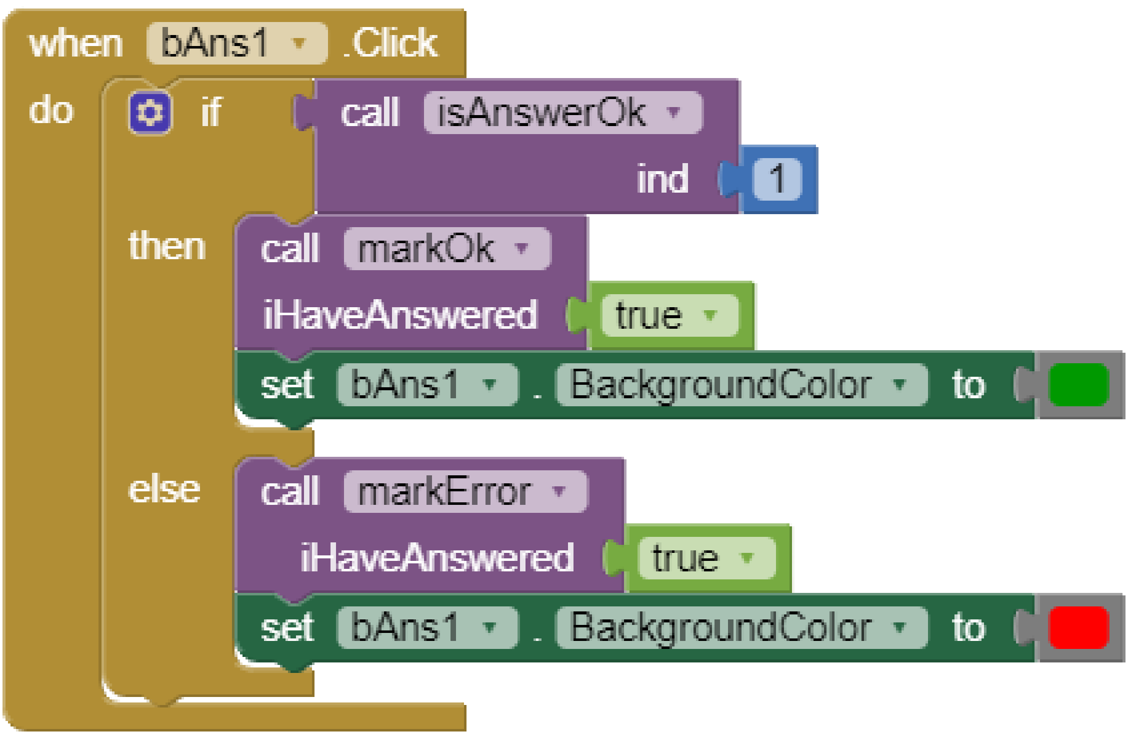
```
to render_answers answers
do  set global LIST_POSSIBLE_ANSWERS to  get answers
    if  length of list list  get answers  ≥  1
    then  set bAns1 . Visible to  true
          set bAns1 . BackgroundColor to  □
          set bAns1 . Text to  select list item list  get answers
                                              index  1
    else  set bAns1 . Visible to  false
    if  length of list list  get answers  ≥  2
    then  set bAns2 . Visible to  true
          set bAns2 . BackgroundColor to  □
          set bAns2 . Text to  select list item list  get answers
                                              index  2
    else  set bAns2 . Visible to  false
    if  length of list list  get answers  ≥  3
    then  set bAns3 . Visible to  true
          set bAns3 . BackgroundColor to  □
          set bAns3 . Text to  select list item list  get answers
                                              index  3
    else  set bAns3 . Visible to  false
    if  length of list list  get answers  ≥  4
    then  set bAns4 . Visible to  true
          set bAns4 . BackgroundColor to  □
          set bAns4 . Text to  select list item list  get answers
                                              index  4
    else  set bAns4 . Visible to  false
```



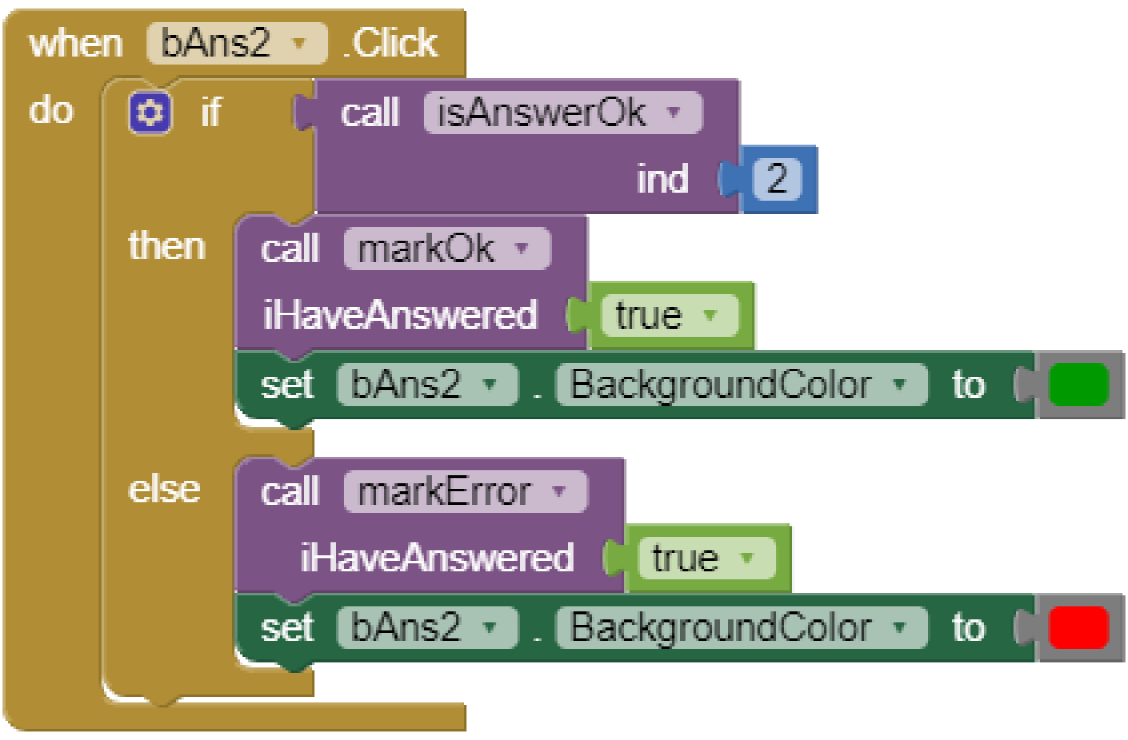
```
to sendMsg2Clients msg
do  if  blueServer . IsConnected
    then  call blueServer .SendText
                           text  get msg

to sendMsg2Server msg
do              blueClient          IsConnected
                 blueClient
                                              msg

to showQuestion item
do  if  is empty  get item
    then  initialize local ind to  call getIndQuestion2Answer
          in  if  get ind > 0
              then  set item to  select list item  list  get global ALL_QUESTIONS
                                                  index  get ind
          if  get global IS_SERVER
          then  call sendMsg2Clients
                             msg  get item
    set global ANSWER to  get value for key  " t "
                               in dictionary  get item
                           or if not found  " not found "
    set txtQuestion . Text to  get value for key  " q "
                                    in dictionary  get item
                                or if not found  " not found "
    ?  if  is key in dictionary? key  " a "
                          dictionary  get item
       then  set item to  make a dictionary  key " t "  value  if  compare texts  " T "  =  get value for key  " a "
                                                                                              in dictionary  get item
                                                                                          or if not found  " not found "
                                                                     then  " Verdadero "
                                                                     else  " Falso "
                                             key " f "  value  make a list  if  compare texts  " T "  =  get value for key  " a "
                                                                                                           in dictionary  get item
                                                                                                       or if not found  " not found "
                                                                                  then  " Falso "
                                                                                  else  " Verdadero "
    initialize local answers to  copy list  list  get value for key  " f "
                                                       in dictionary  get item
                                                   or if not found  " not found "
    in  add items to list  list  get answers
                           item  get value for key  " t "
                                      in dictionary  get item
                                  or if not found  " not found "
        set answers to  call randomize_list
                                      in_list  get answers
        call render_answers
                   answers  get answers
            debug
                                  # answers :
                                                       answers
                       false
        debug
                     Show question
                  false
```



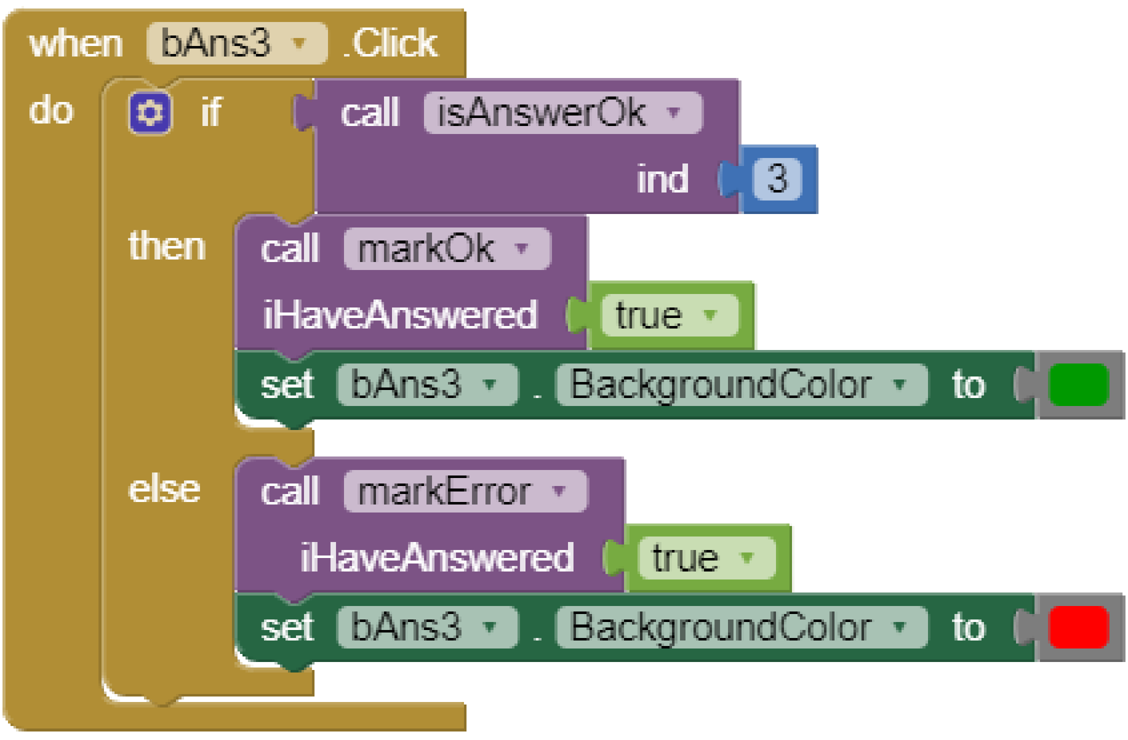
```
when blueServer .ConnectionAccepted
do  call debug
             msg  " I am the server "
          delete  false
    set global IS_SERVER to true
    set global SINGLE_PLAYER to false
    call sendMsg2Clients
                     msg  get global DEF_SERVER_CHOSEN
```



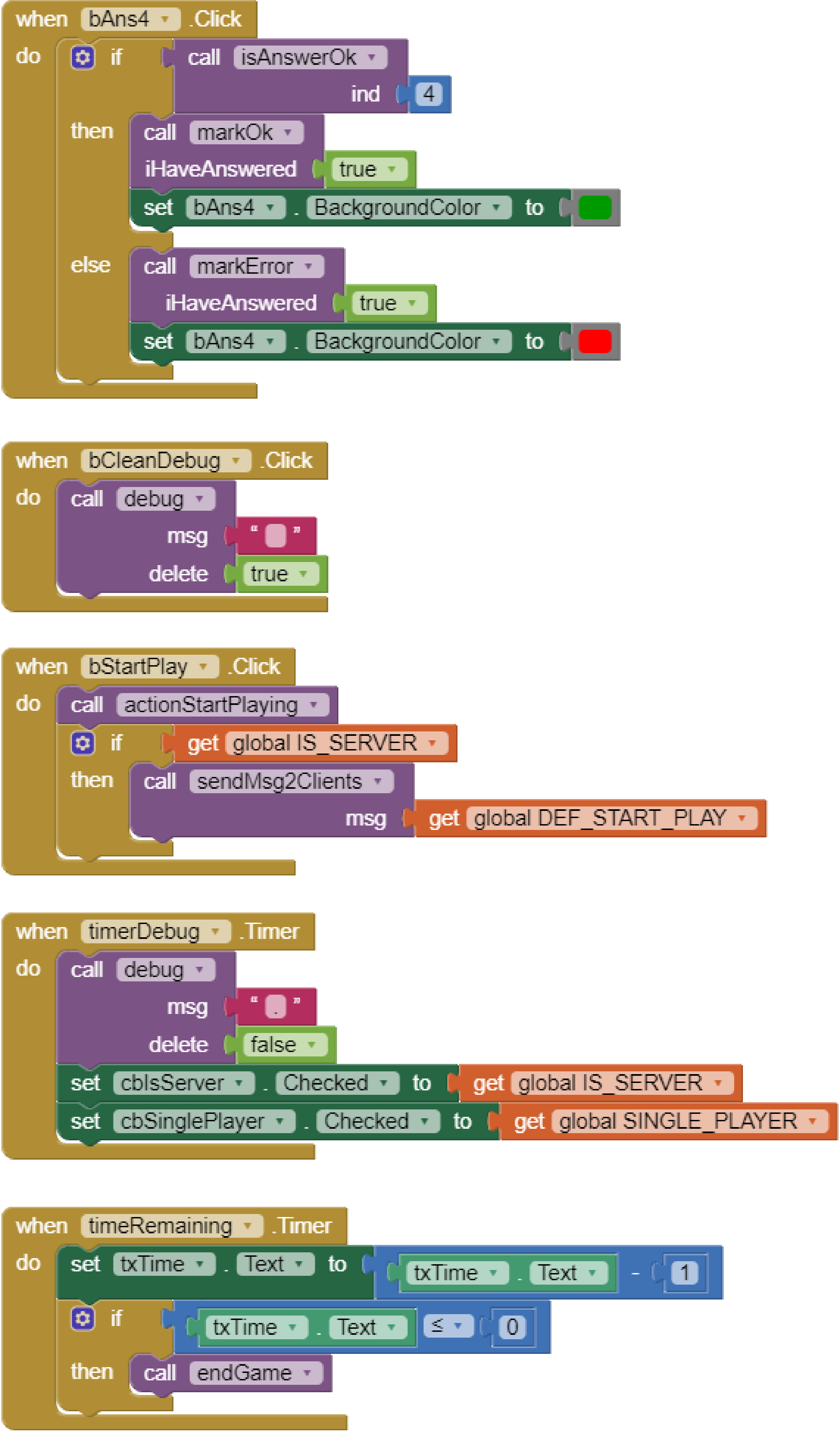
```
when bAns1 .Click
do  if   call isAnswerOk
                      ind  1
    then  call markOk
          iHaveAnswered  true
          set bAns1 . BackgroundColor to 🟩
    else  call markError
            iHaveAnswered  true
          set bAns1 . BackgroundColor to 🟥
```



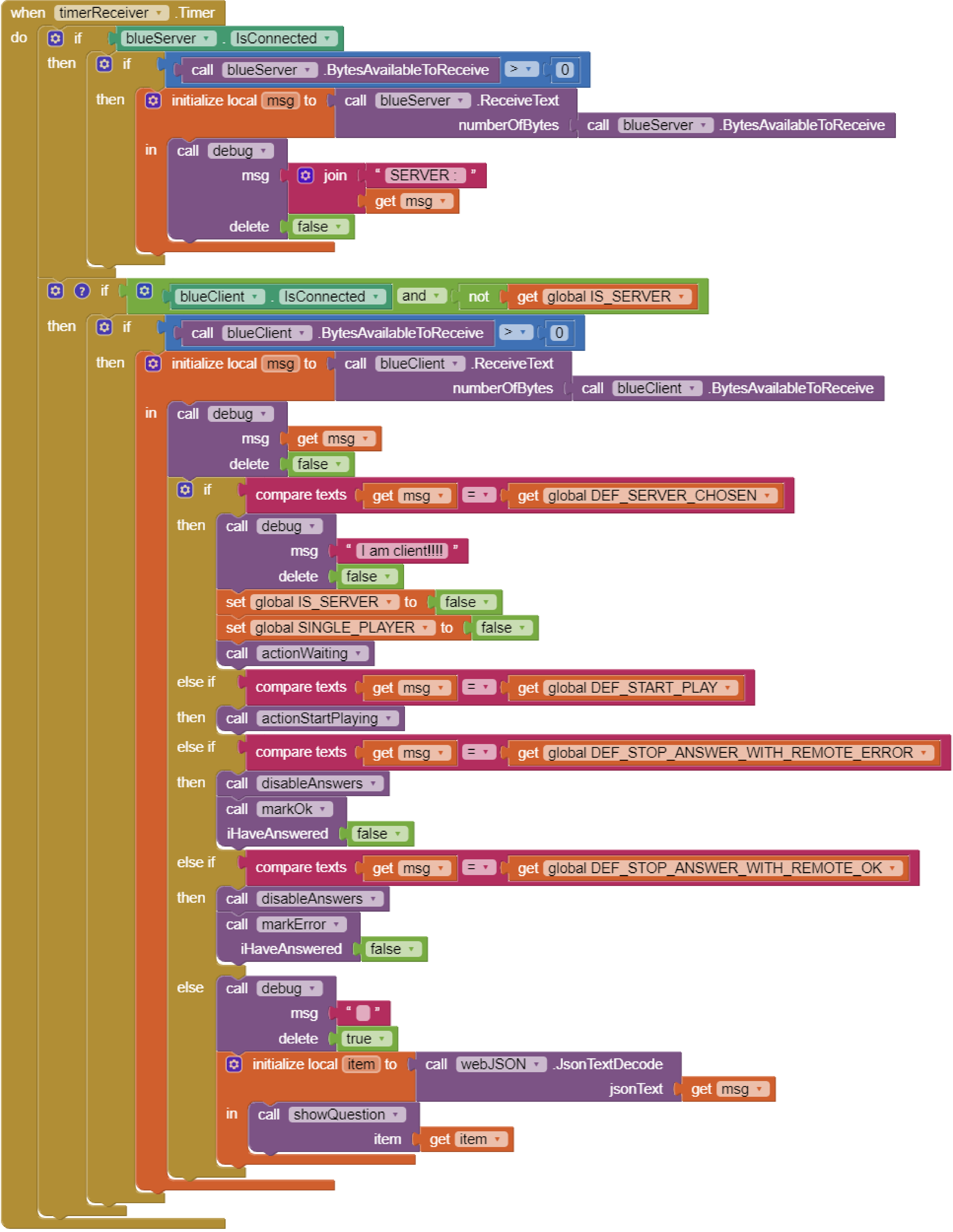
```
when bAns2 .Click
do  if   call isAnswerOk
                      ind  2
    then  call markOk
          iHaveAnswered  true
          set bAns2 . BackgroundColor to 🟩
    else  call markError
            iHaveAnswered  true
          set bAns2 . BackgroundColor to 🟥
```



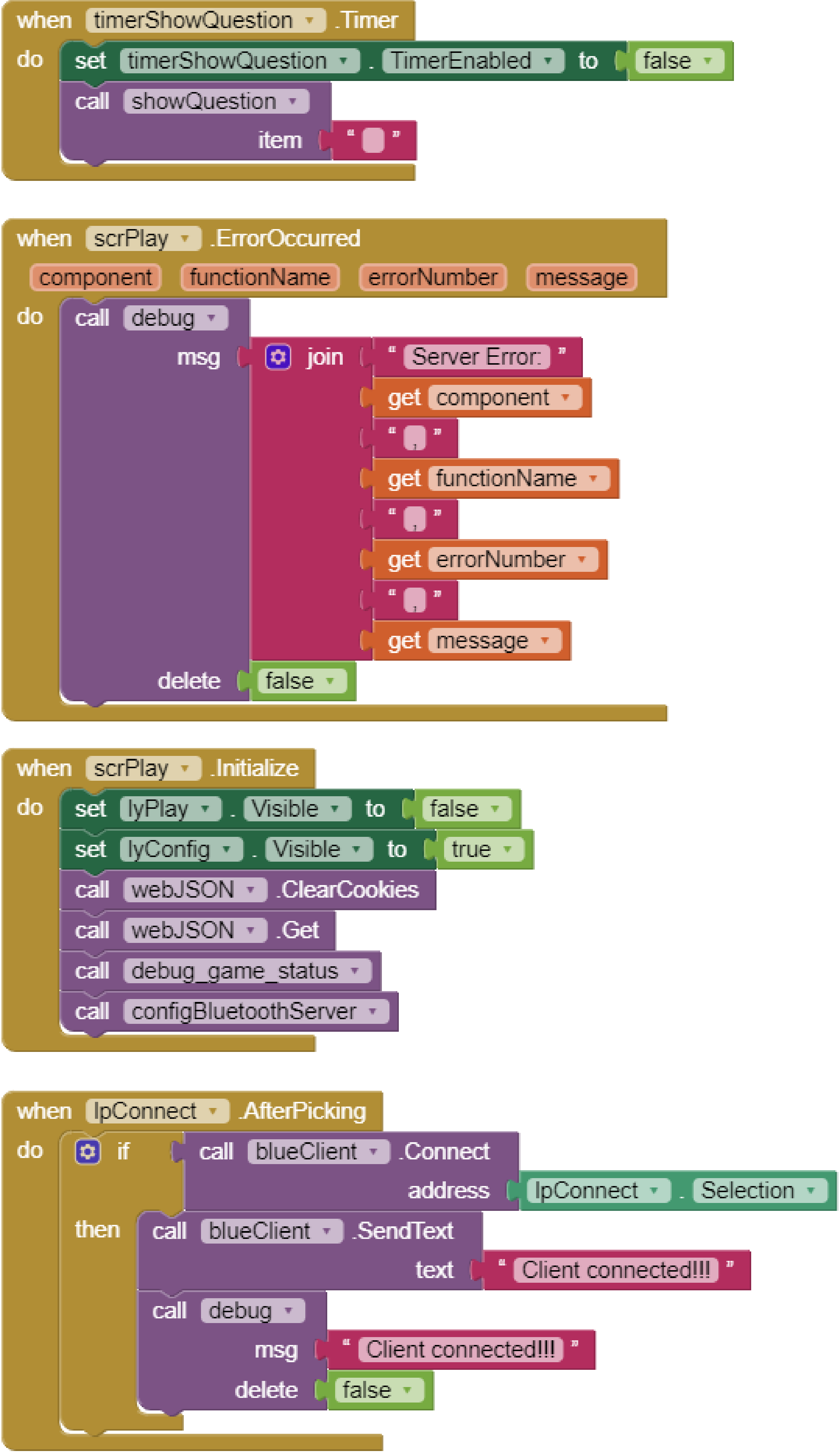
```
when bAns3 .Click
do  if   call isAnswerOk
                      ind  3
    then  call markOk
          iHaveAnswered  true
          set bAns3 . BackgroundColor to 🟩
    else  call markError
            iHaveAnswered  true
          set bAns3 . BackgroundColor to 🟥
```



```
when bAns4 .Click
do  if    call isAnswerOk
                     ind  4
    then  call markOk
            iHaveAnswered  true
          set bAns4 . BackgroundColor to 🟩
    else  call markError
            iHaveAnswered  true
          set bAns4 . BackgroundColor to 🟥

when bCleanDebug .Click
do  call debug
           msg  " "
        delete  true

when bStartPlay .Click
do  call actionStartPlaying
    if    get global IS_SERVER
    then  call sendMsg2Clients
                        msg  get global DEF_START_PLAY

when timerDebug .Timer
do  call debug
           msg  " . "
        delete  false
    set cbIsServer . Checked to  get global IS_SERVER
    set cbSinglePlayer . Checked to  get global SINGLE_PLAYER

when timeRemaining .Timer
do  set txTime . Text to  txTime . Text - 1
    if    txTime . Text ≤ 0
    then  call endGame
```



```
when timerReceiver .Timer
do  if blueServer . IsConnected
    then  if call blueServer .BytesAvailableToReceive > 0
          then  initialize local msg to  call blueServer .ReceiveText
                                              numberOfBytes  call blueServer .BytesAvailableToReceive
                in  call debug
                         msg     join  " SERVER : "
                                       get msg
                         delete  false
    if  blueClient . IsConnected  and  not  get global IS_SERVER
    then  if call blueClient .BytesAvailableToReceive > 0
          then  initialize local msg to  call blueClient .ReceiveText
                                              numberOfBytes  call blueClient .BytesAvailableToReceive
                in  call debug
                         msg     get msg
                         delete  false
                    if  compare texts  get msg  =  get global DEF_SERVER_CHOSEN
                    then  call debug
                               msg     " I am client!!!! "
                               delete  false
                          set global IS_SERVER to false
                          set global SINGLE_PLAYER to false
                          call actionWaiting
                    else if  compare texts  get msg  =  get global DEF_START_PLAY
                    then  call actionStartPlaying
                    else if  compare texts  get msg  =  get global DEF_STOP_ANSWER_WITH_REMOTE_ERROR
                    then  call disableAnswers
                          call markOk
                               iHaveAnswered  false
                    else if  compare texts  get msg  =  get global DEF_STOP_ANSWER_WITH_REMOTE_OK
                    then  call disableAnswers
                          call markError
                               iHaveAnswered  false
                    else  call debug
                               msg     " "
                               delete  true
                          initialize local item to  call webJSON .JsonTextDecode
                                                         jsonText  get msg
                          in  call showQuestion
                                   item  get item
```



```
when timerShowQuestion .Timer
do  set timerShowQuestion . TimerEnabled to false
    call showQuestion
                  item " "

when scrPlay .ErrorOccurred
  component  functionName  errorNumber  message
do  call debug
         msg   join  " Server Error: "
                     get component
                     " , "
                     get functionName
                     " , "
                     get errorNumber
                     " , "
                     get message
      delete  false

when scrPlay .Initialize
do  set lyPlay . Visible to false
    set lyConfig . Visible to true
    call webJSON .ClearCookies
    call webJSON .Get
    call debug_game_status
    call configBluetoothServer

when lpConnect .AfterPicking
do  if    call blueClient .Connect
                       address  lpConnect . Selection
    then  call blueClient .SendText
                          text " Client connected!!! "
          call debug
                msg " Client connected!!! "
             delete false
```