Challenge 2 : Agent communication

For Agent communication there was send-message (msg) function provided. But act-r-experiment.lisp needs to be updated and also geomates.lisp for sending raw format of msg->rect or msg->disc

Solution : solution to this problem was found by updating act-r-experiment.lisp and geomates.listp

Send.message(msg) is updated so that it sends m key first and then message content

```lisp
(defun send-message (msg)
  "sends a message (anything printable, but should be an s-expression)"
  (ensure-connection)
  (when *gstream*
    (format t "Raw message received: ~s~%" msg)  ; ~s will print it with readability preserved
    ;; First send the 'm' character (ASCII 109) which is the message command
    (write-char #\m *gstream*)
    ;; Then write the message itself in a way that can be read by the Lisp reader
    (write msg :stream *gstream* :readably t :pretty nil)
    ;; Ensure the message is sent immediately
    (finish-output *gstream*)))
```

respond-to-key-press (model key) function is updated for msg->rect and msg->disc messages.this message will be added in visicon

```lisp
                          value (polygon  player-info )
                          player-type ,*player-type*)))

    (loop for (what . attributes) in updated-scene do
      (case what
        (:msg->rect (setf *message* attributes)
              (when *message*
              ;; Add message to visicon as a special feature
              (add-visicon-features `(isa (text-feature text)
                               screen-x 10
                               screen-y 10
                               value (text "message")
                               message ,*message*
                               sender "rect"))))

        (:msg->disc (setf *message* attributes)
                (when *message*
                  ;; Add message to visicon as a special feature
                  (add-visicon-features `(isa (text-feature text)
                                 screen-x 10
                                 screen-y 10
                                 value (text "message")
                                 message ,*message*
                                 sender "disc"))))
        (:platform (destructuring-bind (x1 y1 x2 y2) attributes
                    (add-visicon-features `(isa (polygon-feature polygon)
                               screen-x ,(* 0.5 (+ x1 x2))
                               screen-y ,(* 0.5 (+ y1 y2))
                               value (polygon "platform")
                               height ,(abs (- y2 y1))
                               width  ,(abs (- x2 x1))
                               color black regular (true nil) sides (nil 4)))))
```

Player info will be continuously update in visicon so that in model we can read it

Then geomates.lisp is updated from this

```
;; send current scene to anyone listening
(let* ((*print-pretty* nil)
    (current-scene (format nil "((:RECT ~,2f ~,2f ~,2f ~,2f ~,4f ~d)(:DISC ~,2f ~,2f ~,2f ~d)~a~a~{~w~}~{~w~})"
            rect-pos-x rect-pos-y rect-width rect-height rect-rotation diamonds-rect
            disc-pos-x disc-pos-y +disc-radius+ diamonds-disc
            (if message-from-disc
                (list :msg->rect message-from-disc)
                "")
            (if message-from-rect
                (list :msg->disc message-from-rect)
                "")
            diamonds platforms)))
```

To this so that (:msg->rect (:abc :cde 10 20) is not formatted to (msg->rect (abc cde 10 20)

```
;; send current scene to anyone listening
(let* ((*print-pretty* nil)
    (current-scene (format nil "((:RECT ~,2f ~,2f ~,2f ~,2f ~,4f ~d)(:DISC ~,2f ~,2f ~,2f ~d)~s~s~{~w~}~{~w~})"
            rect-pos-x rect-pos-y rect-width rect-height rect-rotation diamonds-rect
            disc-pos-x disc-pos-y +disc-radius+ diamonds-disc
            (if message-from-disc
                (list :msg->rect message-from-disc)
                "")
            (if message-from-rect
                (list :msg->disc message-from-rect)
                "")
            diamonds platforms)))
```

Working code for this can be found on the following location

https://github.com/SandeshGavhane/DeepSeek_ICA_Agent/blob/main/geomates/DeepSeek Agent_Version6_MessageSending.lisp

https://github.com/SandeshGavhane/DeepSeek_ICA_Agent/blob/main/geomates/act-r-experiment.lisp

https://github.com/SandeshGavhane/DeepSeek_ICA_Agent/blob/main/geomates/geomates.lisp