;; Auto-generated. Do not edit!

(when (boundp 'robotnik\_msgs\_new::SetElevatorAction)

(if (not (find-package "ROBOTNIK\_MSGS\_NEW"))

(make-package "ROBOTNIK\_MSGS\_NEW"))

(shadow 'SetElevatorAction (find-package "ROBOTNIK\_MSGS\_NEW")))

(unless (find-package "ROBOTNIK\_MSGS\_NEW::SETELEVATORACTION")

(make-package "ROBOTNIK\_MSGS\_NEW::SETELEVATORACTION"))

(in-package "ROS")

;;//! \htmlinclude SetElevatorAction.msg.html

(defclass robotnik\_msgs\_new::SetElevatorAction

:super ros::object

:slots (\_action\_goal \_action\_result \_action\_feedback ))

(defmethod robotnik\_msgs\_new::SetElevatorAction

(:init

(&key

((:action\_goal \_\_action\_goal) (instance robotnik\_msgs\_new::SetElevatorActionGoal :init))

((:action\_result \_\_action\_result) (instance robotnik\_msgs\_new::SetElevatorActionResult :init))

((:action\_feedback \_\_action\_feedback) (instance robotnik\_msgs\_new::SetElevatorActionFeedback :init))

)

(send-super :init)

(setq \_action\_goal \_\_action\_goal)

(setq \_action\_result \_\_action\_result)

(setq \_action\_feedback \_\_action\_feedback)

self)

(:action\_goal

(&rest \_\_action\_goal)

(if (keywordp (car \_\_action\_goal))

(send\* \_action\_goal \_\_action\_goal)

(progn

(if \_\_action\_goal (setq \_action\_goal (car \_\_action\_goal)))

\_action\_goal)))

(:action\_result

(&rest \_\_action\_result)

(if (keywordp (car \_\_action\_result))

(send\* \_action\_result \_\_action\_result)

(progn

(if \_\_action\_result (setq \_action\_result (car \_\_action\_result)))

\_action\_result)))

(:action\_feedback

(&rest \_\_action\_feedback)

(if (keywordp (car \_\_action\_feedback))

(send\* \_action\_feedback \_\_action\_feedback)

(progn

(if \_\_action\_feedback (setq \_action\_feedback (car \_\_action\_feedback)))

\_action\_feedback)))

(:serialization-length

()

(+

;; robotnik\_msgs\_new/SetElevatorActionGoal \_action\_goal

(send \_action\_goal :serialization-length)

;; robotnik\_msgs\_new/SetElevatorActionResult \_action\_result

(send \_action\_result :serialization-length)

;; robotnik\_msgs\_new/SetElevatorActionFeedback \_action\_feedback

(send \_action\_feedback :serialization-length)

))

(:serialize

(&optional strm)

(let ((s (if strm strm

(make-string-output-stream (send self :serialization-length)))))

;; robotnik\_msgs\_new/SetElevatorActionGoal \_action\_goal

(send \_action\_goal :serialize s)

;; robotnik\_msgs\_new/SetElevatorActionResult \_action\_result

(send \_action\_result :serialize s)

;; robotnik\_msgs\_new/SetElevatorActionFeedback \_action\_feedback

(send \_action\_feedback :serialize s)

;;

(if (null strm) (get-output-stream-string s))))

(:deserialize

(buf &optional (ptr- 0))

;; robotnik\_msgs\_new/SetElevatorActionGoal \_action\_goal

(send \_action\_goal :deserialize buf ptr-) (incf ptr- (send \_action\_goal :serialization-length))

;; robotnik\_msgs\_new/SetElevatorActionResult \_action\_result

(send \_action\_result :deserialize buf ptr-) (incf ptr- (send \_action\_result :serialization-length))

;; robotnik\_msgs\_new/SetElevatorActionFeedback \_action\_feedback

(send \_action\_feedback :deserialize buf ptr-) (incf ptr- (send \_action\_feedback :serialization-length))

;;

self)

)

(setf (get robotnik\_msgs\_new::SetElevatorAction :md5sum-) "201432bcb5eaa69b9ce4956b570dda0a")

(setf (get robotnik\_msgs\_new::SetElevatorAction :datatype-) "robotnik\_msgs\_new/SetElevatorAction")

(setf (get robotnik\_msgs\_new::SetElevatorAction :definition-)

"# ====== DO NOT MODIFY! AUTOGENERATED FROM AN ACTION DEFINITION ======

SetElevatorActionGoal action\_goal

SetElevatorActionResult action\_result

SetElevatorActionFeedback action\_feedback

================================================================================

MSG: robotnik\_msgs\_new/SetElevatorActionGoal

# ====== DO NOT MODIFY! AUTOGENERATED FROM AN ACTION DEFINITION ======

Header header

actionlib\_msgs/GoalID goal\_id

SetElevatorGoal goal

================================================================================

MSG: std\_msgs/Header

# Standard metadata for higher-level stamped data types.

# This is generally used to communicate timestamped data

# in a particular coordinate frame.

#

# sequence ID: consecutively increasing ID

uint32 seq

#Two-integer timestamp that is expressed as:

# \* stamp.sec: seconds (stamp\_secs) since epoch (in Python the variable is called 'secs')

# \* stamp.nsec: nanoseconds since stamp\_secs (in Python the variable is called 'nsecs')

# time-handling sugar is provided by the client library

time stamp

#Frame this data is associated with

string frame\_id

================================================================================

MSG: actionlib\_msgs/GoalID

# The stamp should store the time at which this goal was requested.

# It is used by an action server when it tries to preempt all

# goals that were requested before a certain time

time stamp

# The id provides a way to associate feedback and

# result message with specific goal requests. The id

# specified must be unique.

string id

================================================================================

MSG: robotnik\_msgs\_new/SetElevatorGoal

# ====== DO NOT MODIFY! AUTOGENERATED FROM AN ACTION DEFINITION ======

robotnik\_msgs\_new/ElevatorAction action

================================================================================

MSG: robotnik\_msgs\_new/ElevatorAction

int32 RAISE=1

int32 LOWER=-1

int32 STOP=0

int32 NO\_ACTION=1000

int32 action

# speed, height for future applications

================================================================================

MSG: robotnik\_msgs\_new/SetElevatorActionResult

# ====== DO NOT MODIFY! AUTOGENERATED FROM AN ACTION DEFINITION ======

Header header

actionlib\_msgs/GoalStatus status

SetElevatorResult result

================================================================================

MSG: actionlib\_msgs/GoalStatus

GoalID goal\_id

uint8 status

uint8 PENDING = 0 # The goal has yet to be processed by the action server

uint8 ACTIVE = 1 # The goal is currently being processed by the action server

uint8 PREEMPTED = 2 # The goal received a cancel request after it started executing

# and has since completed its execution (Terminal State)

uint8 SUCCEEDED = 3 # The goal was achieved successfully by the action server (Terminal State)

uint8 ABORTED = 4 # The goal was aborted during execution by the action server due

# to some failure (Terminal State)

uint8 REJECTED = 5 # The goal was rejected by the action server without being processed,

# because the goal was unattainable or invalid (Terminal State)

uint8 PREEMPTING = 6 # The goal received a cancel request after it started executing

# and has not yet completed execution

uint8 RECALLING = 7 # The goal received a cancel request before it started executing,

# but the action server has not yet confirmed that the goal is canceled

uint8 RECALLED = 8 # The goal received a cancel request before it started executing

# and was successfully cancelled (Terminal State)

uint8 LOST = 9 # An action client can determine that a goal is LOST. This should not be

# sent over the wire by an action server

#Allow for the user to associate a string with GoalStatus for debugging

string text

================================================================================

MSG: robotnik\_msgs\_new/SetElevatorResult

# ====== DO NOT MODIFY! AUTOGENERATED FROM AN ACTION DEFINITION ======

bool result

robotnik\_msgs\_new/ElevatorStatus status

================================================================================

MSG: robotnik\_msgs\_new/ElevatorStatus

# state

string RAISING=raising

string LOWERING=lowering

string IDLE=idle

string ERROR\_G\_IO=error\_getting\_io

string ERROR\_S\_IO=error\_setting\_io

string ERROR\_TIMEOUT=error\_timeout\_in\_action

# position

string UP=up

string DOWN=down

string UNKNOWN=unknown

# IDLE, RAISING, LOWERING

string state

# UP, DOWN, UNKNOWN

string position

float32 height

================================================================================

MSG: robotnik\_msgs\_new/SetElevatorActionFeedback

# ====== DO NOT MODIFY! AUTOGENERATED FROM AN ACTION DEFINITION ======

Header header

actionlib\_msgs/GoalStatus status

SetElevatorFeedback feedback

================================================================================

MSG: robotnik\_msgs\_new/SetElevatorFeedback

# ====== DO NOT MODIFY! AUTOGENERATED FROM AN ACTION DEFINITION ======

robotnik\_msgs\_new/ElevatorStatus status

")

(provide :robotnik\_msgs\_new/SetElevatorAction "201432bcb5eaa69b9ce4956b570dda0a")