;; Auto-generated. Do not edit!

(when (boundp 'robotnik\_msgs\_new::set\_ptz)

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

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

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

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

(make-package "ROBOTNIK\_MSGS\_NEW::SET\_PTZ"))

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

(make-package "ROBOTNIK\_MSGS\_NEW::SET\_PTZREQUEST"))

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

(make-package "ROBOTNIK\_MSGS\_NEW::SET\_PTZRESPONSE"))

(in-package "ROS")

(defclass robotnik\_msgs\_new::set\_ptzRequest

:super ros::object

:slots (\_pan \_tilt \_zoom \_relative ))

(defmethod robotnik\_msgs\_new::set\_ptzRequest

(:init

(&key

((:pan \_\_pan) 0.0)

((:tilt \_\_tilt) 0.0)

((:zoom \_\_zoom) 0.0)

((:relative \_\_relative) nil)

)

(send-super :init)

(setq \_pan (float \_\_pan))

(setq \_tilt (float \_\_tilt))

(setq \_zoom (float \_\_zoom))

(setq \_relative \_\_relative)

self)

(:pan

(&optional \_\_pan)

(if \_\_pan (setq \_pan \_\_pan)) \_pan)

(:tilt

(&optional \_\_tilt)

(if \_\_tilt (setq \_tilt \_\_tilt)) \_tilt)

(:zoom

(&optional \_\_zoom)

(if \_\_zoom (setq \_zoom \_\_zoom)) \_zoom)

(:relative

(&optional \_\_relative)

(if \_\_relative (setq \_relative \_\_relative)) \_relative)

(:serialization-length

()

(+

;; float32 \_pan

4

;; float32 \_tilt

4

;; float32 \_zoom

4

;; bool \_relative

1

))

(:serialize

(&optional strm)

(let ((s (if strm strm

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

;; float32 \_pan

(sys::poke \_pan (send s :buffer) (send s :count) :float) (incf (stream-count s) 4)

;; float32 \_tilt

(sys::poke \_tilt (send s :buffer) (send s :count) :float) (incf (stream-count s) 4)

;; float32 \_zoom

(sys::poke \_zoom (send s :buffer) (send s :count) :float) (incf (stream-count s) 4)

;; bool \_relative

(if \_relative (write-byte -1 s) (write-byte 0 s))

;;

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

(:deserialize

(buf &optional (ptr- 0))

;; float32 \_pan

(setq \_pan (sys::peek buf ptr- :float)) (incf ptr- 4)

;; float32 \_tilt

(setq \_tilt (sys::peek buf ptr- :float)) (incf ptr- 4)

;; float32 \_zoom

(setq \_zoom (sys::peek buf ptr- :float)) (incf ptr- 4)

;; bool \_relative

(setq \_relative (not (= 0 (sys::peek buf ptr- :char)))) (incf ptr- 1)

;;

self)

)

(defclass robotnik\_msgs\_new::set\_ptzResponse

:super ros::object

:slots (\_ret ))

(defmethod robotnik\_msgs\_new::set\_ptzResponse

(:init

(&key

((:ret \_\_ret) nil)

)

(send-super :init)

(setq \_ret \_\_ret)

self)

(:ret

(&optional \_\_ret)

(if \_\_ret (setq \_ret \_\_ret)) \_ret)

(:serialization-length

()

(+

;; bool \_ret

1

))

(:serialize

(&optional strm)

(let ((s (if strm strm

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

;; bool \_ret

(if \_ret (write-byte -1 s) (write-byte 0 s))

;;

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

(:deserialize

(buf &optional (ptr- 0))

;; bool \_ret

(setq \_ret (not (= 0 (sys::peek buf ptr- :char)))) (incf ptr- 1)

;;

self)

)

(defclass robotnik\_msgs\_new::set\_ptz

:super ros::object

:slots ())

(setf (get robotnik\_msgs\_new::set\_ptz :md5sum-) "b0b0a0b023ba1e7f5ff6a9d48ee0689b")

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

(setf (get robotnik\_msgs\_new::set\_ptz :request) robotnik\_msgs\_new::set\_ptzRequest)

(setf (get robotnik\_msgs\_new::set\_ptz :response) robotnik\_msgs\_new::set\_ptzResponse)

(defmethod robotnik\_msgs\_new::set\_ptzRequest

(:response () (instance robotnik\_msgs\_new::set\_ptzResponse :init)))

(setf (get robotnik\_msgs\_new::set\_ptzRequest :md5sum-) "b0b0a0b023ba1e7f5ff6a9d48ee0689b")

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

(setf (get robotnik\_msgs\_new::set\_ptzRequest :definition-)

"# PAN value

float32 pan

# Tilt value

float32 tilt

# Zoom value

float32 zoom

# Flag for relative movements

bool relative

---

bool ret

")

(setf (get robotnik\_msgs\_new::set\_ptzResponse :md5sum-) "b0b0a0b023ba1e7f5ff6a9d48ee0689b")

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

(setf (get robotnik\_msgs\_new::set\_ptzResponse :definition-)

"# PAN value

float32 pan

# Tilt value

float32 tilt

# Zoom value

float32 zoom

# Flag for relative movements

bool relative

---

bool ret

")

(provide :robotnik\_msgs\_new/set\_ptz "b0b0a0b023ba1e7f5ff6a9d48ee0689b")