;; Auto-generated. Do not edit!

(when (boundp 'robotnik\_msgs\_new::set\_CartesianEuler\_pose)

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

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

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

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

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

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

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

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

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

(in-package "ROS")

(defclass robotnik\_msgs\_new::set\_CartesianEuler\_poseRequest

:super ros::object

:slots (\_x \_y \_z \_A \_B \_C ))

(defmethod robotnik\_msgs\_new::set\_CartesianEuler\_poseRequest

(:init

(&key

((:x \_\_x) 0.0)

((:y \_\_y) 0.0)

((:z \_\_z) 0.0)

((:A \_\_A) 0.0)

((:B \_\_B) 0.0)

((:C \_\_C) 0.0)

)

(send-super :init)

(setq \_x (float \_\_x))

(setq \_y (float \_\_y))

(setq \_z (float \_\_z))

(setq \_A (float \_\_A))

(setq \_B (float \_\_B))

(setq \_C (float \_\_C))

self)

(:x

(&optional \_\_x)

(if \_\_x (setq \_x \_\_x)) \_x)

(:y

(&optional \_\_y)

(if \_\_y (setq \_y \_\_y)) \_y)

(:z

(&optional \_\_z)

(if \_\_z (setq \_z \_\_z)) \_z)

(:A

(&optional \_\_A)

(if \_\_A (setq \_A \_\_A)) \_A)

(:B

(&optional \_\_B)

(if \_\_B (setq \_B \_\_B)) \_B)

(:C

(&optional \_\_C)

(if \_\_C (setq \_C \_\_C)) \_C)

(:serialization-length

()

(+

;; float32 \_x

4

;; float32 \_y

4

;; float32 \_z

4

;; float32 \_A

4

;; float32 \_B

4

;; float32 \_C

4

))

(:serialize

(&optional strm)

(let ((s (if strm strm

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

;; float32 \_x

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

;; float32 \_y

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

;; float32 \_z

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

;; float32 \_A

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

;; float32 \_B

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

;; float32 \_C

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

;;

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

(:deserialize

(buf &optional (ptr- 0))

;; float32 \_x

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

;; float32 \_y

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

;; float32 \_z

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

;; float32 \_A

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

;; float32 \_B

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

;; float32 \_C

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

;;

self)

)

(defclass robotnik\_msgs\_new::set\_CartesianEuler\_poseResponse

:super ros::object

:slots (\_ret ))

(defmethod robotnik\_msgs\_new::set\_CartesianEuler\_poseResponse

(: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\_CartesianEuler\_pose

:super ros::object

:slots ())

(setf (get robotnik\_msgs\_new::set\_CartesianEuler\_pose :md5sum-) "2519f4d60ca7104c2905597455df82b0")

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

(setf (get robotnik\_msgs\_new::set\_CartesianEuler\_pose :request) robotnik\_msgs\_new::set\_CartesianEuler\_poseRequest)

(setf (get robotnik\_msgs\_new::set\_CartesianEuler\_pose :response) robotnik\_msgs\_new::set\_CartesianEuler\_poseResponse)

(defmethod robotnik\_msgs\_new::set\_CartesianEuler\_poseRequest

(:response () (instance robotnik\_msgs\_new::set\_CartesianEuler\_poseResponse :init)))

(setf (get robotnik\_msgs\_new::set\_CartesianEuler\_poseRequest :md5sum-) "2519f4d60ca7104c2905597455df82b0")

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

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

"# New robot X position (mm)

float32 x

# New robot Y position (mm)

float32 y

# New robot Z position (mm)

float32 z

# New robot Orientation (grad)

float32 A

# New robot Orientation (grad)

float32 B

# New robot Orientation (grad)

float32 C

---

bool ret

")

(setf (get robotnik\_msgs\_new::set\_CartesianEuler\_poseResponse :md5sum-) "2519f4d60ca7104c2905597455df82b0")

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

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

"# New robot X position (mm)

float32 x

# New robot Y position (mm)

float32 y

# New robot Z position (mm)

float32 z

# New robot Orientation (grad)

float32 A

# New robot Orientation (grad)

float32 B

# New robot Orientation (grad)

float32 C

---

bool ret

")

(provide :robotnik\_msgs\_new/set\_CartesianEuler\_pose "2519f4d60ca7104c2905597455df82b0")