;; Auto-generated. Do not edit!

(when (boundp 'gazebo\_msgs\_new::SetJointProperties)

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

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

(shadow 'SetJointProperties (find-package "GAZEBO\_MSGS\_NEW")))

(unless (find-package "GAZEBO\_MSGS\_NEW::SETJOINTPROPERTIES")

(make-package "GAZEBO\_MSGS\_NEW::SETJOINTPROPERTIES"))

(unless (find-package "GAZEBO\_MSGS\_NEW::SETJOINTPROPERTIESREQUEST")

(make-package "GAZEBO\_MSGS\_NEW::SETJOINTPROPERTIESREQUEST"))

(unless (find-package "GAZEBO\_MSGS\_NEW::SETJOINTPROPERTIESRESPONSE")

(make-package "GAZEBO\_MSGS\_NEW::SETJOINTPROPERTIESRESPONSE"))

(in-package "ROS")

(defclass gazebo\_msgs\_new::SetJointPropertiesRequest

:super ros::object

:slots (\_joint\_name \_ode\_joint\_config ))

(defmethod gazebo\_msgs\_new::SetJointPropertiesRequest

(:init

(&key

((:joint\_name \_\_joint\_name) "")

((:ode\_joint\_config \_\_ode\_joint\_config) (instance gazebo\_msgs\_new::ODEJointProperties :init))

)

(send-super :init)

(setq \_joint\_name (string \_\_joint\_name))

(setq \_ode\_joint\_config \_\_ode\_joint\_config)

self)

(:joint\_name

(&optional \_\_joint\_name)

(if \_\_joint\_name (setq \_joint\_name \_\_joint\_name)) \_joint\_name)

(:ode\_joint\_config

(&rest \_\_ode\_joint\_config)

(if (keywordp (car \_\_ode\_joint\_config))

(send\* \_ode\_joint\_config \_\_ode\_joint\_config)

(progn

(if \_\_ode\_joint\_config (setq \_ode\_joint\_config (car \_\_ode\_joint\_config)))

\_ode\_joint\_config)))

(:serialization-length

()

(+

;; string \_joint\_name

4 (length \_joint\_name)

;; gazebo\_msgs\_new/ODEJointProperties \_ode\_joint\_config

(send \_ode\_joint\_config :serialization-length)

))

(:serialize

(&optional strm)

(let ((s (if strm strm

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

;; string \_joint\_name

(write-long (length \_joint\_name) s) (princ \_joint\_name s)

;; gazebo\_msgs\_new/ODEJointProperties \_ode\_joint\_config

(send \_ode\_joint\_config :serialize s)

;;

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

(:deserialize

(buf &optional (ptr- 0))

;; string \_joint\_name

(let (n) (setq n (sys::peek buf ptr- :integer)) (incf ptr- 4) (setq \_joint\_name (subseq buf ptr- (+ ptr- n))) (incf ptr- n))

;; gazebo\_msgs\_new/ODEJointProperties \_ode\_joint\_config

(send \_ode\_joint\_config :deserialize buf ptr-) (incf ptr- (send \_ode\_joint\_config :serialization-length))

;;

self)

)

(defclass gazebo\_msgs\_new::SetJointPropertiesResponse

:super ros::object

:slots (\_success \_status\_message ))

(defmethod gazebo\_msgs\_new::SetJointPropertiesResponse

(:init

(&key

((:success \_\_success) nil)

((:status\_message \_\_status\_message) "")

)

(send-super :init)

(setq \_success \_\_success)

(setq \_status\_message (string \_\_status\_message))

self)

(:success

(&optional \_\_success)

(if \_\_success (setq \_success \_\_success)) \_success)

(:status\_message

(&optional \_\_status\_message)

(if \_\_status\_message (setq \_status\_message \_\_status\_message)) \_status\_message)

(:serialization-length

()

(+

;; bool \_success

1

;; string \_status\_message

4 (length \_status\_message)

))

(:serialize

(&optional strm)

(let ((s (if strm strm

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

;; bool \_success

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

;; string \_status\_message

(write-long (length \_status\_message) s) (princ \_status\_message s)

;;

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

(:deserialize

(buf &optional (ptr- 0))

;; bool \_success

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

;; string \_status\_message

(let (n) (setq n (sys::peek buf ptr- :integer)) (incf ptr- 4) (setq \_status\_message (subseq buf ptr- (+ ptr- n))) (incf ptr- n))

;;

self)

)

(defclass gazebo\_msgs\_new::SetJointProperties

:super ros::object

:slots ())

(setf (get gazebo\_msgs\_new::SetJointProperties :md5sum-) "15a76e921b1410c698d82782ea5fa370")

(setf (get gazebo\_msgs\_new::SetJointProperties :datatype-) "gazebo\_msgs\_new/SetJointProperties")

(setf (get gazebo\_msgs\_new::SetJointProperties :request) gazebo\_msgs\_new::SetJointPropertiesRequest)

(setf (get gazebo\_msgs\_new::SetJointProperties :response) gazebo\_msgs\_new::SetJointPropertiesResponse)

(defmethod gazebo\_msgs\_new::SetJointPropertiesRequest

(:response () (instance gazebo\_msgs\_new::SetJointPropertiesResponse :init)))

(setf (get gazebo\_msgs\_new::SetJointPropertiesRequest :md5sum-) "15a76e921b1410c698d82782ea5fa370")

(setf (get gazebo\_msgs\_new::SetJointPropertiesRequest :datatype-) "gazebo\_msgs\_new/SetJointPropertiesRequest")

(setf (get gazebo\_msgs\_new::SetJointPropertiesRequest :definition-)

"string joint\_name # name of joint

gazebo\_msgs\_new/ODEJointProperties ode\_joint\_config # access to ODE joint dynamics properties

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

MSG: gazebo\_msgs\_new/ODEJointProperties

# access to low level joint properties, change these at your own risk

float64[] damping # joint damping

float64[] hiStop # joint limit

float64[] loStop # joint limit

float64[] erp # set joint erp

float64[] cfm # set joint cfm

float64[] stop\_erp # set joint erp for joint limit \"contact\" joint

float64[] stop\_cfm # set joint cfm for joint limit \"contact\" joint

float64[] fudge\_factor # joint fudge\_factor applied at limits, see ODE manual for info.

float64[] fmax # ode joint param fmax

float64[] vel # ode joint param vel

---

bool success # return true if get successful

string status\_message # comments if available

")

(setf (get gazebo\_msgs\_new::SetJointPropertiesResponse :md5sum-) "15a76e921b1410c698d82782ea5fa370")

(setf (get gazebo\_msgs\_new::SetJointPropertiesResponse :datatype-) "gazebo\_msgs\_new/SetJointPropertiesResponse")

(setf (get gazebo\_msgs\_new::SetJointPropertiesResponse :definition-)

"string joint\_name # name of joint

gazebo\_msgs\_new/ODEJointProperties ode\_joint\_config # access to ODE joint dynamics properties

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

MSG: gazebo\_msgs\_new/ODEJointProperties

# access to low level joint properties, change these at your own risk

float64[] damping # joint damping

float64[] hiStop # joint limit

float64[] loStop # joint limit

float64[] erp # set joint erp

float64[] cfm # set joint cfm

float64[] stop\_erp # set joint erp for joint limit \"contact\" joint

float64[] stop\_cfm # set joint cfm for joint limit \"contact\" joint

float64[] fudge\_factor # joint fudge\_factor applied at limits, see ODE manual for info.

float64[] fmax # ode joint param fmax

float64[] vel # ode joint param vel

---

bool success # return true if get successful

string status\_message # comments if available

")

(provide :gazebo\_msgs\_new/SetJointProperties "15a76e921b1410c698d82782ea5fa370")