;; Auto-generated. Do not edit!

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

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

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

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

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

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

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

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

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

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

(in-package "ROS")

(if (not (find-package "GEOMETRY\_MSGS"))

(ros::roseus-add-msgs "geometry\_msgs"))

(defclass gazebo\_msgs\_new::GetLinkPropertiesRequest

:super ros::object

:slots (\_link\_name ))

(defmethod gazebo\_msgs\_new::GetLinkPropertiesRequest

(:init

(&key

((:link\_name \_\_link\_name) "")

)

(send-super :init)

(setq \_link\_name (string \_\_link\_name))

self)

(:link\_name

(&optional \_\_link\_name)

(if \_\_link\_name (setq \_link\_name \_\_link\_name)) \_link\_name)

(:serialization-length

()

(+

;; string \_link\_name

4 (length \_link\_name)

))

(:serialize

(&optional strm)

(let ((s (if strm strm

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

;; string \_link\_name

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

;;

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

(:deserialize

(buf &optional (ptr- 0))

;; string \_link\_name

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

;;

self)

)

(defclass gazebo\_msgs\_new::GetLinkPropertiesResponse

:super ros::object

:slots (\_com \_gravity\_mode \_mass \_ixx \_ixy \_ixz \_iyy \_iyz \_izz \_success \_status\_message ))

(defmethod gazebo\_msgs\_new::GetLinkPropertiesResponse

(:init

(&key

((:com \_\_com) (instance geometry\_msgs::Pose :init))

((:gravity\_mode \_\_gravity\_mode) nil)

((:mass \_\_mass) 0.0)

((:ixx \_\_ixx) 0.0)

((:ixy \_\_ixy) 0.0)

((:ixz \_\_ixz) 0.0)

((:iyy \_\_iyy) 0.0)

((:iyz \_\_iyz) 0.0)

((:izz \_\_izz) 0.0)

((:success \_\_success) nil)

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

)

(send-super :init)

(setq \_com \_\_com)

(setq \_gravity\_mode \_\_gravity\_mode)

(setq \_mass (float \_\_mass))

(setq \_ixx (float \_\_ixx))

(setq \_ixy (float \_\_ixy))

(setq \_ixz (float \_\_ixz))

(setq \_iyy (float \_\_iyy))

(setq \_iyz (float \_\_iyz))

(setq \_izz (float \_\_izz))

(setq \_success \_\_success)

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

self)

(:com

(&rest \_\_com)

(if (keywordp (car \_\_com))

(send\* \_com \_\_com)

(progn

(if \_\_com (setq \_com (car \_\_com)))

\_com)))

(:gravity\_mode

(&optional \_\_gravity\_mode)

(if \_\_gravity\_mode (setq \_gravity\_mode \_\_gravity\_mode)) \_gravity\_mode)

(:mass

(&optional \_\_mass)

(if \_\_mass (setq \_mass \_\_mass)) \_mass)

(:ixx

(&optional \_\_ixx)

(if \_\_ixx (setq \_ixx \_\_ixx)) \_ixx)

(:ixy

(&optional \_\_ixy)

(if \_\_ixy (setq \_ixy \_\_ixy)) \_ixy)

(:ixz

(&optional \_\_ixz)

(if \_\_ixz (setq \_ixz \_\_ixz)) \_ixz)

(:iyy

(&optional \_\_iyy)

(if \_\_iyy (setq \_iyy \_\_iyy)) \_iyy)

(:iyz

(&optional \_\_iyz)

(if \_\_iyz (setq \_iyz \_\_iyz)) \_iyz)

(:izz

(&optional \_\_izz)

(if \_\_izz (setq \_izz \_\_izz)) \_izz)

(: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

()

(+

;; geometry\_msgs/Pose \_com

(send \_com :serialization-length)

;; bool \_gravity\_mode

1

;; float64 \_mass

8

;; float64 \_ixx

8

;; float64 \_ixy

8

;; float64 \_ixz

8

;; float64 \_iyy

8

;; float64 \_iyz

8

;; float64 \_izz

8

;; 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)))))

;; geometry\_msgs/Pose \_com

(send \_com :serialize s)

;; bool \_gravity\_mode

(if \_gravity\_mode (write-byte -1 s) (write-byte 0 s))

;; float64 \_mass

(sys::poke \_mass (send s :buffer) (send s :count) :double) (incf (stream-count s) 8)

;; float64 \_ixx

(sys::poke \_ixx (send s :buffer) (send s :count) :double) (incf (stream-count s) 8)

;; float64 \_ixy

(sys::poke \_ixy (send s :buffer) (send s :count) :double) (incf (stream-count s) 8)

;; float64 \_ixz

(sys::poke \_ixz (send s :buffer) (send s :count) :double) (incf (stream-count s) 8)

;; float64 \_iyy

(sys::poke \_iyy (send s :buffer) (send s :count) :double) (incf (stream-count s) 8)

;; float64 \_iyz

(sys::poke \_iyz (send s :buffer) (send s :count) :double) (incf (stream-count s) 8)

;; float64 \_izz

(sys::poke \_izz (send s :buffer) (send s :count) :double) (incf (stream-count s) 8)

;; 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))

;; geometry\_msgs/Pose \_com

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

;; bool \_gravity\_mode

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

;; float64 \_mass

(setq \_mass (sys::peek buf ptr- :double)) (incf ptr- 8)

;; float64 \_ixx

(setq \_ixx (sys::peek buf ptr- :double)) (incf ptr- 8)

;; float64 \_ixy

(setq \_ixy (sys::peek buf ptr- :double)) (incf ptr- 8)

;; float64 \_ixz

(setq \_ixz (sys::peek buf ptr- :double)) (incf ptr- 8)

;; float64 \_iyy

(setq \_iyy (sys::peek buf ptr- :double)) (incf ptr- 8)

;; float64 \_iyz

(setq \_iyz (sys::peek buf ptr- :double)) (incf ptr- 8)

;; float64 \_izz

(setq \_izz (sys::peek buf ptr- :double)) (incf ptr- 8)

;; 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::GetLinkProperties

:super ros::object

:slots ())

(setf (get gazebo\_msgs\_new::GetLinkProperties :md5sum-) "0e06a70386d0ee3fb880c02f23fcd821")

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

(setf (get gazebo\_msgs\_new::GetLinkProperties :request) gazebo\_msgs\_new::GetLinkPropertiesRequest)

(setf (get gazebo\_msgs\_new::GetLinkProperties :response) gazebo\_msgs\_new::GetLinkPropertiesResponse)

(defmethod gazebo\_msgs\_new::GetLinkPropertiesRequest

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

(setf (get gazebo\_msgs\_new::GetLinkPropertiesRequest :md5sum-) "0e06a70386d0ee3fb880c02f23fcd821")

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

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

"string link\_name # name of link

# link names are prefixed by model name, e.g. pr2::base\_link

---

geometry\_msgs/Pose com # center of mass location in link frame

# and orientation of the moment of inertias

# relative to the link frame

bool gravity\_mode # set gravity mode on/off

float64 mass # linear mass of link

float64 ixx # moment of inertia

float64 ixy # moment of inertia

float64 ixz # moment of inertia

float64 iyy # moment of inertia

float64 iyz # moment of inertia

float64 izz # moment of inertia

bool success # return true if get info is successful

string status\_message # comments if available

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

MSG: geometry\_msgs/Pose

# A representation of pose in free space, composed of position and orientation.

Point position

Quaternion orientation

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

MSG: geometry\_msgs/Point

# This contains the position of a point in free space

float64 x

float64 y

float64 z

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

MSG: geometry\_msgs/Quaternion

# This represents an orientation in free space in quaternion form.

float64 x

float64 y

float64 z

float64 w

")

(setf (get gazebo\_msgs\_new::GetLinkPropertiesResponse :md5sum-) "0e06a70386d0ee3fb880c02f23fcd821")

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

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

"string link\_name # name of link

# link names are prefixed by model name, e.g. pr2::base\_link

---

geometry\_msgs/Pose com # center of mass location in link frame

# and orientation of the moment of inertias

# relative to the link frame

bool gravity\_mode # set gravity mode on/off

float64 mass # linear mass of link

float64 ixx # moment of inertia

float64 ixy # moment of inertia

float64 ixz # moment of inertia

float64 iyy # moment of inertia

float64 iyz # moment of inertia

float64 izz # moment of inertia

bool success # return true if get info is successful

string status\_message # comments if available

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

MSG: geometry\_msgs/Pose

# A representation of pose in free space, composed of position and orientation.

Point position

Quaternion orientation

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

MSG: geometry\_msgs/Point

# This contains the position of a point in free space

float64 x

float64 y

float64 z

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

MSG: geometry\_msgs/Quaternion

# This represents an orientation in free space in quaternion form.

float64 x

float64 y

float64 z

float64 w

")

(provide :gazebo\_msgs\_new/GetLinkProperties "0e06a70386d0ee3fb880c02f23fcd821")