;; Auto-generated. Do not edit!

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

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

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

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

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

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

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

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

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

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

(in-package "ROS")

(defclass robotnik\_msgs\_new::InsertTaskRequest

:super ros::object

:slots (\_id\_submission \_description\_task \_datatime\_start ))

(defmethod robotnik\_msgs\_new::InsertTaskRequest

(:init

(&key

((:id\_submission \_\_id\_submission) 0)

((:description\_task \_\_description\_task) "")

((:datatime\_start \_\_datatime\_start) "")

)

(send-super :init)

(setq \_id\_submission (round \_\_id\_submission))

(setq \_description\_task (string \_\_description\_task))

(setq \_datatime\_start (string \_\_datatime\_start))

self)

(:id\_submission

(&optional \_\_id\_submission)

(if \_\_id\_submission (setq \_id\_submission \_\_id\_submission)) \_id\_submission)

(:description\_task

(&optional \_\_description\_task)

(if \_\_description\_task (setq \_description\_task \_\_description\_task)) \_description\_task)

(:datatime\_start

(&optional \_\_datatime\_start)

(if \_\_datatime\_start (setq \_datatime\_start \_\_datatime\_start)) \_datatime\_start)

(:serialization-length

()

(+

;; int32 \_id\_submission

4

;; string \_description\_task

4 (length \_description\_task)

;; string \_datatime\_start

4 (length \_datatime\_start)

))

(:serialize

(&optional strm)

(let ((s (if strm strm

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

;; int32 \_id\_submission

(write-long \_id\_submission s)

;; string \_description\_task

(write-long (length \_description\_task) s) (princ \_description\_task s)

;; string \_datatime\_start

(write-long (length \_datatime\_start) s) (princ \_datatime\_start s)

;;

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

(:deserialize

(buf &optional (ptr- 0))

;; int32 \_id\_submission

(setq \_id\_submission (sys::peek buf ptr- :integer)) (incf ptr- 4)

;; string \_description\_task

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

;; string \_datatime\_start

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

;;

self)

)

(defclass robotnik\_msgs\_new::InsertTaskResponse

:super ros::object

:slots (\_success \_msg ))

(defmethod robotnik\_msgs\_new::InsertTaskResponse

(:init

(&key

((:success \_\_success) nil)

((:msg \_\_msg) "")

)

(send-super :init)

(setq \_success \_\_success)

(setq \_msg (string \_\_msg))

self)

(:success

(&optional \_\_success)

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

(:msg

(&optional \_\_msg)

(if \_\_msg (setq \_msg \_\_msg)) \_msg)

(:serialization-length

()

(+

;; bool \_success

1

;; string \_msg

4 (length \_msg)

))

(: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 \_msg

(write-long (length \_msg) s) (princ \_msg 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 \_msg

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

;;

self)

)

(defclass robotnik\_msgs\_new::InsertTask

:super ros::object

:slots ())

(setf (get robotnik\_msgs\_new::InsertTask :md5sum-) "36ad6270fb0855e3101179d36268dd90")

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

(setf (get robotnik\_msgs\_new::InsertTask :request) robotnik\_msgs\_new::InsertTaskRequest)

(setf (get robotnik\_msgs\_new::InsertTask :response) robotnik\_msgs\_new::InsertTaskResponse)

(defmethod robotnik\_msgs\_new::InsertTaskRequest

(:response () (instance robotnik\_msgs\_new::InsertTaskResponse :init)))

(setf (get robotnik\_msgs\_new::InsertTaskRequest :md5sum-) "36ad6270fb0855e3101179d36268dd90")

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

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

"int32 id\_submission

string description\_task

string datatime\_start

---

bool success

string msg

")

(setf (get robotnik\_msgs\_new::InsertTaskResponse :md5sum-) "36ad6270fb0855e3101179d36268dd90")

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

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

"int32 id\_submission

string description\_task

string datatime\_start

---

bool success

string msg

")

(provide :robotnik\_msgs\_new/InsertTask "36ad6270fb0855e3101179d36268dd90")