

--: {port: []: (:smiley:) <- **Transport Operations Command Line Library** -> muse/docs/lib/port.md  
--:neutral\_face: port: *CLL for **book** and **port** commands assessing and clearing player inventory for specified teleport.* -> port

--:# Partially consumed inventory items add to the energy **bank** used to decrease a future porting cost in a session.

--:# Default player inventory items are considered for **book** and **port** (porting) operations.

--:> port.item: *Short inventory item name for booking* -> "**iron**"|"copper"|"gold"|"redstone"

--:# Rather arbitrarily, height for porting includes a player standing on a turtle

--:# **Support and test functions for **book** and **port** operations.**

--:+ *Assess player inventory considered for energy available to do work.*

--: port.inventory(costToPay: #:, orderedItems: port.order, playerInventory: [port.item]: #:smiley: -> *Testing* -> ":"[], #:, #:

--:+ *Returns payment inventory used for **costToPay**, excess energy to bank, inventory energy used*

--:> port.order: *Order in which inventory items will be considered for booking* -> **port.item[]**

--: port.available(set: [port.item]: #:smiley: -> *For Testing: mock player inventory* -> [**port.item**]: #:

--: port.book(name: ":", label: ":", from: place, to: place, span: #:, ordering: port.order) -> *Testing* -> **booking: ":"**

--:+ *Booking (string) names the **range** entry that will be used to specify a **port** operation and how to pay for it.*

--:+ *Return needed inventory for booking and bankable if > 0*

--:# **Booking from place spanning area to place, offering items for port operation**

--:- book name label from to span? item??... -> *Spanned range with (default) items as properties; return cost less bank.*

--:# Save booking as a **range** with **span** and offered or default ordering of items as **range** properties

--:# **Port operation consuming player inventory offered in booking.**

--:- port booking -> *As provided in booking, consume player inventory to teleport entities from one area to another.*

--:# **If this is in-game, do the port and if successful, pay for it by clearing counts of selected items**

--: port.op(commands: ":"[]) -> *Command Line Interface for teleport book and trip* -> ":" &: