**Inventory Class Design Notes**

**========================**

Inventory refers to items that are immediately accessible by the player's character at all times.

Storage refers to items that are stored in the Storage Chest in the player's house.

The Inventory class will support a variable number of items. Additional slots can be purchased or earned by achievements (victories or levels)

Inventory class data:

Items – implemented as a dictionary (item name is the key, item count is the data).

Inventory class methods:

* Take – pick up an item (from the local environment) and add it to the inventory (if space is available). “take” can be abbreviated as “t”, “ta”, or “tak”. Issuing the take command without any argument will result in attempting to take all available items (same as issuing the command as “take all”). If an argument is given, it can be an item type (“weapon” or “weapons”, “armor”, “potion” or “potions”, “spell” or “spells”, “key” or “keys”, “ring” or “rings”, “trophy” or “trophies”, “gold”, or “all”), or it can be a partial name, such as, “dragon” or “rusty”. If it is a recognized type, the system will attempt to take only items of that type. If it is a partial name, the system will find all items that contain that text in the item name. When any of these operations results in an ambiguous list (more than one item meets the criteria, the system will list all matched items and issue a prompt to allow the player to select the desired item (“all” will be the last choice on the selection list). If there are more than one copy of a unique item, the system will prompt the player to specify how many should be taken (“all” will be a valid response to take all copies).
  + take all :: takes all items available
  + take :: same as take all
  + take sword :: takes only a sword; if multiple are available, selection list is presented
  + t bandages 4 :: takes up to 4 bandages; warning given if less than four are found
* Drop – drop an item (will then be shown in the local environment)
  + drop all :: drops all inventory items after prompt: “Are you sure?”
  + drop :: displays a selection list of all items, choose items to drop
  + drop boots :: drops boots from inventory, not boots that are currently being worn
  + d weapons :: drops weapons from inventory, not weapon that is currently wielded
  + d spells :: drops items from the spells inventory
  + d potions :: drops items from the potions inventory
* List – display a list of current inventory items
  + inv all :: displays all inventory items in a list
  + inventory :: same as inventory all
  + in sword :: displays all inventory items with “sword” in the name
  + i armor :: displays all armor items in inventory (and worn)
  + i weapons :: displays all weapon items in inventory (and wielded)

Take – pick up an item (from the local environment) and add it to the inventory (if space is available).

* “take” can be abbreviated as “t”, “ta”, or “tak”.
* Issuing the take command without an argument will result in attempting to take all available items (same as issuing the command, “take all”).
* If an argument is given, it can be an item type:
  + “weapon” or “weapons”
  + “armor”
  + “potion” or “potions”
  + “spell” or “spells”
  + “key” or “keys”
  + “ring” or “rings”
  + “trophy” or “trophies”
  + “gold”
  + “all”
  + or it can be a partial name, such as, “dragon” or “rusty”.
* If it is a recognized type, the system will attempt to take only items of that type.
* If it is a partial name (anything other than a recognized type), the system will find all items that contain the specified text in the item name.
* When any take operation results in an ambiguous list (more than one item meets the criteria), the system will list all matched items and issue a prompt to allow the player to select the desired item.
  + “all” will be the last choice on the selection list.
* If there is more than one copy of a unique item, the system will prompt the player to specify how many should be taken.
  + “all” will be a valid response to take all copies.