

# card\_game\_kit

Created by: peterdownie.ails to auto generate, working on a fixcom

Software Version: 2.41k

Greater Code Name: Atomic

Lesser Code Name: Neutron

Service Version: 0.01

## Purpose

- \* CREATED BY: Peter Downie of peterdownie.com
- \* Card class for making card programs!
- \* Seperated into 4 catagories
- \* CARDS are the deck!
- \* Pile is for piles
- \* Hands are for players hands, they are basically piles but allow for better distinction.
- \* Counter is use for keeping track of scores or tricks.

## lowerText

Creator did not override lowerText

## technicalSpecifications

Was not implemented!

## Main Methods

saveData

restoreData

generateCards\_standard

generateCards\_special

shuffleCards

seeRemainingCards

dealCard

put\_cards\_on\_top\_of\_deck

put\_cards\_on\_bottom\_of\_deck

bury\_cards\_into\_random\_part\_of\_deck

makePile  
makeHand  
addCardsToHand  
addCardsToPile  
removeCardsFromHand  
removeCardsFromPile  
getAllCardsInHand  
getAllCardsInPile  
get\_top\_card  
make\_counter  
set\_counter  
modify\_counter  
get\_counterValue

## Method Details

### **saveData**

| Save the data for later use in function restoreData

### **restoreData**

| *Parameter #0 [ <required> \$savedData ]*

| Restore Saved Data Produced by function saveData

### **generateCards\_standard**

| *Parameter #0 [ <required> \$number\_of\_decks ]*

| Generates standard playing cards with X number of decks.

### **generateCards\_special**

| *Parameter #0 [ <required> \$number\_of\_decks ]*

| *Parameter #1 [ <required> \$cardValues ]*

| *Parameter #2 [ <optional> \$suits = Array ]*

| Generate a custom deck, for example a euchre deck which only uses 9,10,J,Q,K,A

### **shuffleCards**

| Shuffle the cards

### **seeRemainingCards**

\* See how many cards are left in the deck in the current order.

## **dealCard**

*Parameter #0 [ <optional> \$number\_of\_cards = 1 ]*

## **put\_cards\_on\_top\_of\_deck**

*Parameter #0 [ <required> \$cards ]*

\* Puts cards on the top of the deck

\* Does not return errors, must have suit and value in array.

## **put\_cards\_on\_bottom\_of\_deck**

*Parameter #0 [ <required> \$cards ]*

Puts cards onto the bottom of the deck.

Does not return errors, must have suit and value in array.

## **bury\_cards\_into\_random\_part\_of\_deck**

*Parameter #0 [ <required> \$cards ]*

Buries cards into the deck. The cards stay together.

## **makePile**

*Parameter #0 [ <required> \$pileName ]*

Create a new pile.

## **makeHand**

*Parameter #0 [ <required> \$handName ]*

Create a new Hand.

## **addCardsToHand**

*Parameter #0 [ <required> \$handName ]*

*Parameter #1 [ <required> \$cards ]*

Adds cards to a players hand!

## **addCardsToPile**

*Parameter #0 [ <required> \$pileName ]*

*Parameter #1 [ <required> \$cards ]*

\* Add cards to a pile, based on add to a hand with better coding.

## **removeCardsFromHand**

*Parameter #0 [ <required> \$handName ]*

*Parameter #1 [ <required> \$cards ]*

Removes a card from a users hand. Removes multiple of same card only if specified.

## **removeCardsFromPile**

*Parameter #0 [ <required> \$pileName ]*

*Parameter #1 [ <required> \$cards ]*

\* Remove cards by suit and value from a specific pile.

## **getAllCardsInHand**

*Parameter #0 [ <required> \$handName ]*

Get all cards in a users hand.

## **getAllCardsInPile**

*Parameter #0 [ <required> \$pileName ]*

Gets all of the cards from a pile.

## **get\_top\_card**

*Parameter #0 [ <required> \$grouping ]*

*Parameter #1 [ <optional> \$groupName = false ]*

## **make\_counter**

*Parameter #0 [ <required> \$counterName ]*

*Parameter #1 [ <optional> \$default\_value = 0 ]*

Creates a counter and gives it a default value.

## **set\_counter**

*Parameter #0 [ <required> \$counterName ]*

*Parameter #1 [ <required> \$value ]*

Sets a counter to a value

## **modify\_counter**

*Parameter #0 [ <required> \$counterName ]*

| *Parameter #1 [ <optional> \$modifier = 1 ]*

| Adds or subtracts values from a counter.

### **get\_counterValue**

| *Parameter #0 [ <required> \$counterName ]*

| Gets the current value of a counter.

## **Method Technical Specifications**

# **saveData**

No technical specifications found

# **restoreData**

No technical specifications found

# **generateCards\_standard**

No technical specifications found

# **generateCards\_special**

No technical specifications found

# **shuffleCards**

No technical specifications found

# **seeRemainingCards**

No technical specifications found

# **dealCard**

No technical specifications found

# **put\_cards\_on\_top\_of\_deck**

No technical specifications found

# **put\_cards\_on\_bottom\_of\_deck**

No technical specifications found

# **bury\_cards\_into\_random\_part\_of\_deck**

No technical specifications found

# **makePile**

No technical specifications found

# **makeHand**

No technical specifications found

# **addCardsToHand**

No technical specifications found

# **addCardsToPile**

No technical specifications found

# **removeCardsFromHand**

No technical specifications found

# **removeCardsFromPile**

No technical specifications found

# **getAllCardsInHand**

No technical specifications found

# **getAllCardsInPile**

No technical specifications found

# **get\_top\_card**

No technical specifications found

# **make\_counter**

No technical specifications found

# **set\_counter**

No technical specifications found

# **modify\_counter**

No technical specifications found

# **get\_counterValue**

No technical specifications found