

BLACK JACK (poker failed)...

# Features

- User is able to declare money and is able to bet that money
- app holds onto player money value throughout runtime
- [Introduction to Blackjack]
- Will calculate and inform the user how off he is from 21
- Option to keep playing
- draws random cards from a deck
- and displays in game format
- determines win condition

# Cards

```
class Card
  attr_reader :value, :suit

  def initialize(value, suit)
    @value = value
    @suit = suit
  end

  def human_value
    human_value = {
      1 => :ace,
      2 => 2,
      3 => 3,
      4 => 4,
      5 => 5,
      6 => 6,
      7 => 7,
      8 => 8,
      9 => 9,
      10 => 10,
      11 => :jack,
      12 => :queen,
      13 => :king
    }
  end

  def to_s
    "the #{human_value[value]} of #{@suit}"
  end
end
```

# Deck

```
require_relative './card.rb'
class Deck
  SUITS = [:Hearts, :Diamonds, :Clubs, :Spades]

  def initialize
    @cards = SUITS.map{|suit| build(suit)}.flatten.shuffle
  end

  def draw
    @cards.pop
  end

  def build(suit)
    (1..13).map {|value| Card.new(value, suit)}
  end
end
```

# Control - Single Round of BJ(WIP)

```
deck = Deck.new
card = deck.draw
hand = []
hand << card

puts "Dealer: \'Psst. Got money?\' "
money = gets.chomp.to_i
puts "how much this game?"
bet = gets.chomp.to_i
puts hand
begin
  case bet
  when bet > money
    puts "try again bud. not enough!"
  else
    puts "your options are:"
    puts "1. stay"
    puts "2. hit"
    option = gets.chomp
  end
end
```

```
if option == "stay"
  arr = hand.map{|card| card.value}
  total = arr.reduce(&:+)
  puts "Your total is: #{total}"
  if total < 21
    puts "you win #{bet * 2}"
    money += bet * 2
    puts "you now have $#{money} left to play with"
  else total > 21
    puts "lost to nobody."
    money -= bet
    puts "you now have $#{money} left to play with"
  end
elsif option == "hit"
  hand << deck.draw
  puts "cards:"
  hand.each {|card| puts card}
  arr = hand.map{|card| card.value}
  total = arr.reduce(&:+)
  difference = 21 - total
  puts "you are now #{difference} from 21"
else option == 'exit'
  break
end
end until option == "stay"
```

# TODO:

- Gems: tty prompt , Ascii Art
- Computer response - computer player
- Win check
- Loop to play
- Display cards
- Error handling - Invalid input.