**BLACK JACK Game implemented using Python**

ARAVIND KRISHNAKUMAR

[**Aravind.krishnakumar@mavs.uta.edu**](mailto:Aravind.krishnakumar@mavs.uta.edu)

**Project Description:**

Blackjack game was implemented using Python 2.7  
  
**Modules used:**

The program uses two classes **User** and **Dealer**

In User class modules includes

**Deal () –** For the initial deal of cards, the cards are randomly generated using the random library. If the cards include either Aces with 10,Jack,Queen and King any of these cards they are called as **“Blackjack Cards”** and the if user gets these cards they are declared as Winner(without giving the chance for dealer).The generated cards are tracked by maintaining them in a **LIST data structure.**

**Hit ()** – After the initial generation of cards the user has 3 options to choose from **Hit, Stand and Double Down.** If the user selects the hit option these function will be called. The generated card will be appended to the existing LIST. The function will run as long as Hit option is selected. Depending on the summation of the cards the **Aces value will be either as 1 or 11.** The sum of the cards and list of user cards will be displayed as output of selection. If the sum of the cards exceeds the value 21 then the user will **BUSTED** and the credits which had bet will be lost

**Double down () –** When this option is selected, **only one card is generated** for the user and sum of cards of the user and dealer is compared and the winner is declared. If the generated cards value extends more than 21 then the user is busted and the same applies for the Dealer too. **The credits win or lost will be 3/2 times the original credit. Hit ()** module is **reused** for generation of the cards

This option will only be enabled if the both the user cards are same

**Split () –** Both the user cards will be **split into separate lists**. Each separate list will be considered as a separate game where user can select Hit, Stand.This function handles the first set of cards

**Split1 () -** This function handles the second set of split cards

**Splitcardgenerator () –** This function used to generate user cards.

Dealer modules will include

**Ddeal()** – This function is used to generate the initial Dealer cards .Only one card will be displayed to the user and other will be as **Folded.** If the cards include either Ace with 10, Jack, Queen and King any of these cards they are called as **“Blackjack Cards”** the dealer will be declared as the winner.

**Stand()-** If the user selects Stand option then the dealer’s folded card along with the generated cards of dealers will be displayed

The dealer card will be generated if the summation of the cards is **less than 17,**if the dealer sum of cards is more than 17 then cards will be not be generated. If the generated cards, sum of cards value extends 21 then the **dealer is BUSTED (User will be declared as Winner).**

**Cardgenerator () –** Used for generating the Dealer’s cards

**Sumcalc() –** Calculating the summation of the cards

**Checksum()** – Used to check whether the dealer can generate more cards or not using the sum of the cards

**StandSplit()-** The stand function meant when the Split scenario is faced.