Enumerations

Enumerations

Enumerations are used to store a fixed set of values. Unless specified an enumerations will be stored as a zero-based integer value. For example, you might want a direction type that can store one of the values north, south, east or west.

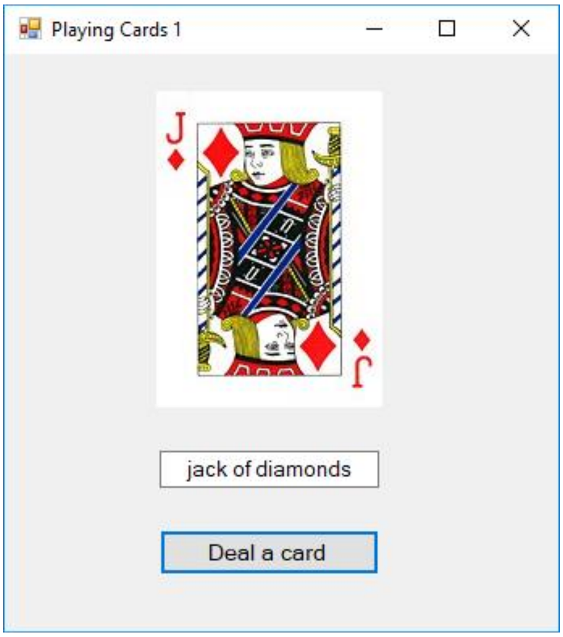
```
public enum eDirection
{
  NORTH = 1,
  SOUTH = 2,
  EAST = 3,
  WEST = 4
}
```

or

```
public enum eDirection
{
  NORTH,
  SOUTH,
  EAST,
  WEST
}
```

Task 1: Playing Cards 1

Write a program to simulate the drawing of a random card from a deck of 52 cards (13 cards for each of the 4 suits). This is a useful utility for building card games like poker or solitaire.



jack of diamonds

1. Create a new project. Put *CardsVersion2* files into the Resources folder. Create enumerations for eSuit and eRank.

2. Place a *Button*, *PictureBox* and a *TextBox* on the *Form*. Set the *Text* property of the *Button* to "Deal a Card" (or equivalent).
3. Create a *Card* class to represent a single playing card. A playing card has a suit and a rank, both of which should be an *enumeration* and an image. The Card class needs a *ToString()* method for writing to the form. This will need to *override* the standard *ToString()* method.

4. Create a *Deck* class which will hold all the possible 52 cards. The constructor will be used to populate the deck. It will also need a *DealACard()* method to choose one card at random and return the chosen *Card*.

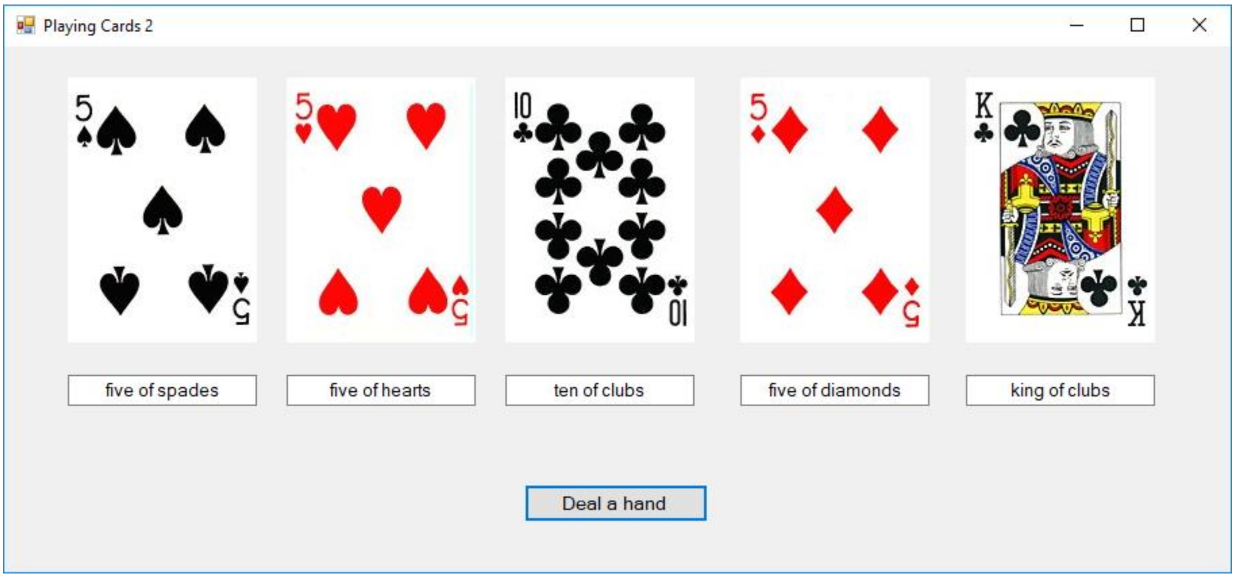
5. In the *Form1 class*, create a *Deck* object (an instance of the *Deck* class).

6. The *button1_Click* handler should ask the deck to call its *DealACard()* method. Remember, this method returns a *Card.* Using that card display the card's image in the PictureBox and write the card's String into the TextBox .

Task 2: Playing Cards 2

******* CHECKPOINT 8 *******

Write a program to simulate the dealing of a hand of cards, that is, five random cards from a deck of cards.



Playing Cards 2

five of spades

five of hearts

ten of clubs

five of diamonds

king of clubs

1. Copy the folder you created for Task1.

2. Place a *Button,* 5 *PictureBoxes* and 5 *TextBoxes* on the *Form*. Set the *Text* property of the *Button* to "Deal a Hand" (or equivalent).

3. In addition to the *Card and Deck* classes, create a *Hand* class to hold the 5 cards (a hand of cards). It will also need a *DealAHand()* method that asks the deck to deal 5 cards from the deck. That is, the deck calls its *DealACard()* method to find a card and the hand stores the card in the cards array in the *Hand* class. Repeat this process 5 times.

4. In the *Form*, make a hand object in the Form, and pass a reference to the deck in the *Hand's* constructor.

5. The *button1_Click* handler should ask the hand to call its *DealAHand()* method. Remember, this method creates a new set of 5 cards. Use the hand to display each card's image in the PictureBox and write each card's String into the TextBox.

6. What is the problem with this implementation?