Solutions: Use to Build 3 Classes for Unit 5 Project - Card Game

1. Fill in the method names for the Card class.

```
public class Card
{
    private String animal;
    private int pic;

    public ___Card___ (){
        pic = 1;
        animal = "anteater";
    }

    public ___Card___ (String a, int p){
        animal = a;
        pic = p;
    }

    public void ___setAnimal___ (String a){
        animal = a;
    }

    public void ___setPic___ (int p){
        pic = p;
    }

    public String getHint (){
        return "It begins with " +
            animal.charAt (0);
    }

    public String getPic (){
        return "animal" + pic + ".png";
    }

    public String ___getAnimal___ (){
        return animal;
    }

    public boolean ___equals___ (Card c){
        if (animal.equals (c.getAnimal ()))
            return true;
        else
            return false;
    }

    public int ___compareTo___ (Card c){
        if (animal.compareTo(c.getAnimal())>0)
            return 1;
        else if (animal.compareTo(c.getAnimal())<0)
            return -1;
        else
            return 0;
    }
}
```

2. Edit the Stack class to hold the Card object

```
public class Deck {
    private int count;
    private ~~Object~~ Card data[];

    public ___Deck___ () {
        count = 0;
    }

    public void push (~~Object~~ Card addMe) {
        data [count] = addMe;
        count++;
    }

    public int size () {
        return count;
    }

    public boolean isFull () {
        return (count == 50);
    }

    public ~~Object~~ Card pop () {
        count--;
        return data [count];
    }

    public ~~Object~~ Card peek () {
        return data [count-1];
    }

    public boolean isEmpty () {
        return count == 0;
    }

    public void clear () {
        count = 0;
    }
}
```

3. Answer true/false about this program.

(F) a. There are 9 code classes.
(T) b. There are 9 picture files. +question
(T) c. The Card class has one instance variable. +question
(T) d. The Deck class is a stack.
(F) e. The AnimalGuessingGame class is an applet.
(T) f. The AnimalGuessingGame class declares a Deck object and a Card object.

Animal Guessing Game 4.9

4. Fill in the arrays with the object information found in the pictures below.

```
//Added to the bottom of the stack class.
public void shuffle ()
{
    //1. set up arrays to hold the card data
    String n[] = {"_anteater_", "_bear_", "_chameleon_", "_duck_",
        "_fox_", "_octopus_", "_seahorse_", "_turtle_"};

    int p[] = {_1_, _2_, _3_, _4_, _5_, _6_, _7_, _8_};

    //2. shuffle the order of the arrays
    for (int i = 0 ; i < 60 ; i++) {
        int a = (int) (Math.random () * n.length);
        int b = (int) (Math.random () * n.length);

        //shuffle the pic numbers
        ___int___   temp = p [a];
        p [a] = p [b];
        p [b] = temp;

        //shuffle the names
        ___String___  temp2 = n [a];
        n [a] = n [b];
        n [b] = temp2;
    }

    //3. make new objects and feed into the stack
    clear ();
    for (int i = 0 ; i < n.length ; i++) {
        Card c = new ___Card___ (n [i], p [i]);
        push (__c__);
    }
}
```

5. Fill in the following.

In the Card class, how many...

(a)	Constructors	2
(b)	Mutators	2
(c)	Accessors	3
(d)	Facilitators	2

In the applet, what are the method calls to do the following?

(a)	Shuffle the deck
(b)	Get the hint
(c)	Get the current picture's name
(d)	Pop a card

Animal Guessing Game

1 - anteater
animal1.png

2 - bear
animal2.png

3 - chameleon
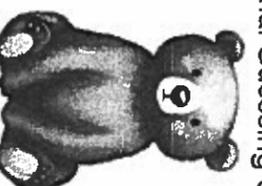animal3.png

4 - duck
animal4.png

5 - fox
animal5.png

6 - octopus
animal6.png

7 - seahorse
animal7.png

8 - turtle
animal8.png

Animal Guessing Game

What is the animal name?
Guess: _____   Guess

You have 0 out of 0 correct.

Get Hint   Shuffle

AnimalGuessingGame running

6. This is the applet that would run the Card and Deck files. Fill in the blanks.

```java
import javax.swing.*; import java.awt.*;
import java.awt.event.*; import java.applet.Applet;

public class AnimalGuessingGame extends Applet implements ActionListener{
    JLabel picture, output, score;
    JTextField guess;
    JButton nextCard, done;
    int right = 0;
    int total = 0;

    //declare Card object & Deck

    Card current = new Card ();

    Deck cardDeck = new Deck ();

    public void init (){

        resize (350, 500);
        JLabel title = new JLabel ("Animal Guessing Game");
        picture = new JLabel (createImageIcon ("blank.png"));

        JLabel pmt = new JLabel ("Guess:");
        guess = new JTextField (8);
        done = new JButton ("Guess");
        done.addActionListener (this);
        done.setActionCommand ("guess");
        done.setEnabled (false);

        output = new JLabel ("First, press new card to begin.");
        score = new JLabel ("You have 000 out of 000 correct..");

        JButton shuffle = new JButton ("Shuffle");
        shuffle.addActionListener (this);
        shuffle.setActionCommand ("shuffle");
        nextCard = new JButton ("Next Card");
        nextCard.addActionListener (this);
        nextCard.setActionCommand ("nextCard");
        JButton getHint = new JButton ("Get Hint");
        getHint.addActionListener (this);
        getHint.setActionCommand ("getHint");

        add (title);    add (picture);    add (output);
        Panel guessLine = new Panel ();
        guessLine.add (pmt);
        guessLine.add (guess);
        guessLine.add (done);
        add (guessLine);
        Panel buttonLine = new Panel ();
        buttonLine.add (nextCard);
        buttonLine.add (getHint);
        buttonLine.add (shuffle);
        add (buttonLine);

        //shuffle the Card Deck so it is ready to begin
        cardDeck . shuffle ();
    }
```



```java
public void actionPerformed (ActionEvent e)
{
    if (e.getActionCommand ().equals ("nextCard")) {

        nextCard .setEnabled (false);
        done .setEnabled (true);

        if (!cardDeck .isEmpty ()) {
            current = cardDeck.pop () ;
            picture .setIcon (
                createImageIcon(current. getPic () ));
            guess .setText ("");
            output .setText ("What is the animal name?");
        }
        else {
            output .setText("No more cards, press shuffle..");
        }
    } //end nextCard
```



```java
    else if (e.getActionCommand ().equals ("guess")) {

        nextCard .setEnabled (true);              //Find answer
        done .setEnabled (false);

        String ans = current.getAnimal () ;       //Find answer
        String theirGuess = guess.getText () ;    //Find guess
        theirGuess = theirGuess.toLowerCase () ;  //Handle typing errors

        if (theirGuess.equals (ans )) {           //if Right
            output .setText ("You got it!");
            right ++;
        }
        else {                                    //if wrong
            output .setText("Sorry, it was " + ans );
        }
        total ++;
        score.setText ("You have " + right + " out of "
                + total + " correct.");

    }//end guess
}//end ap

protected static ImageIcon createImageIcon (String path)
{
    java.net.URL imgURL = AnimalGuessingGame.class.getResource (path);
    if (imgURL != null)
        return new ImageIcon (imgURL);
    else
        return null;
}

}//end applet
```

```java
    else if (e.getActionCommand ().equals ("getHint")) {

        output .setText( current.getHint () );
    } //end getHint

    else if (e.getActionCommand ().equals ("shuffle")) {
        cardDeck.shuffle ();
        total = 0 ;
        right = 0 ;
        guess .setText ("");
        output .setText ("Press new card to begin.");
        picture .setIcon (createImageIcon ("blank.png"));
        score .setText ("You have " + right
                + " of " + total + " correct..");
    }//end shuffle
```


blank.gif

7. Label the global widget & variable names on this diagram:

Labels: picture, output, done, score, guess, nextCard, right, total