CECS 174 – Lecture 24

Writing Classes and Their Methods

Classes – are programmer defined data types. They encapsulate data and methods that are common to the object. Classes are made up of instance variables that store information about the object, and methods to access or alter the object's data. They are usually created in a separate file for better encapsulation and code reuse.

```
public class Dice
{
```

Data Members – information about (or attributes of) the object.

```
     private final int MAX = 6;
     private int diceVal;
```

Constructor – sets the initial values of the data members.

```
     public Dice()
     {
          diceVal = 1;
     }
```

Methods – actions (or operations) the object can perform.

```
     //Rolls the die and returns the result
     public int roll()
     {
          diceVal = (int)(Math.random() * MAX) + 1;
          return diceVal;
     }
```

Accessor – retrieves the value of one of the data members.

```
     public int getDiceVal()
     {
          return diceVal;
     }
```

Mutator – changes the value of one of the data members.

```
     public void setDiceVal( int value ){
          if ( value > 0 && value <= MAX ){
               diceVal = value;
          } else {
               diceVal = 1;
          }
     }
}
```

Objects – are an instance of a class. They are created and used in your program and are manipulated by using the methods defined in the class to access and modify the object's instance variables.

```
public class Game
{
     public static void main (String[] args)
     {

          Dice dice1, dice2;

          dice1 = new Dice();
          dice2 = new Dice();

          dice1.roll();
          dice2.roll();

          System.out.println("Die1="+ dice1.getDiceVal());
          System.out.println("Die2="+ dice2.getDiceVal());

          if( dice1.getDiceVal() > dice2.getDiceVal())
          {
               System.out.println("Player 1 Wins");
          }
          else if(dice2.getDiceVal() > dice1.getDiceVal())
          {
               System.out.println("Player 2 Wins");
          }
          else
          {
               System.out.println("Tie");
          }
     }
```

Adding Methods – we can add functionality to the program by putting extra methods into the Game class. Objects are passed by reference, so if any of the values of an object are changed in a method, the values of the original object are changed as well.

```
     public static void cheat (Dice diceX)
     {
          diceX.setDiceVal(6);
     }
}
```

We can call this method from the main method with:

```
     cheat(dice2);
```