

Conceptos fundamentales de Java

7-1: Clases, objetos y métodos

Actividades practices

Objetivos de la lección:

- Reconocimiento del formato general correcto de una clase
- Creación de un objeto de clase
- Descripción de referencias de objetos
- Creación de métodos de compilación sin errores
- Retorno de un valor a partir de un método
- Uso de parámetros en un método
- Crear una clase controlador y agregar instancias de clases objetos
- Agregar un constructor a una clase
- Aplicar el nuevo operador
- Descripción de la recolección de elementos no utilizados y finalizadores
- Aplicar esta referencia
- Agregar un constructor para inicializar un valor

Vocabulario:

Identifique el término correspondiente a cada una de las definiciones a continuación.

	Plantilla utilizada para realizar objetos Java.
	Palabra clave opcional para tener acceso a los miembros y los métodos de una clase.
	Instancia de una clase.
	Operador utilizado para crear una instancia de una clase.
	Función incorporada en la máquina virtual Java que libera memoria como objetos que ya no son necesarios o que no se hacen referencia.
	Método que cambia el estado de un objeto.
	Método que devuelve información sobre un objeto al programa de llamada.
	Procedimiento (cambia el estado de un objeto) o función (devuelve información sobre un objeto) encapsulado como parte de una clase.
	Verbo utilizado para describir el acto de crear un objeto de clase mediante la palabra clave new (nuevo).
	Proceso de asignar un valor predeterminado a una variable.
	Referencia a un objeto que no se ha instanciado.

	Método opcional llamado justo antes de que el recolector de basura elimina un objeto.
	Nombre de una variable asociada con un objeto.
	Método especial utilizado para crear una instancia de una clase.

Inténtelo/resuélvalo:

1. Cree una forma de clase simple que representa una forma en 2 dimensiones con segmentos de líneas para los bordes. Debe tener la siguiente variable instancia: numSides (int), regular (booleano). Cree al menos dos constructores y métodos captadores y establecedores.
2. Identifique las partes clave de la Clase Java a continuación. Coloque asteriscos junto a todas las variables de instancias. Coloque un casillero junto a cada constructor. Haga un círculo en la firma de los métodos diferentes al método de constructor. Coloque triángulos alrededor de los parámetros. Subraye los tipos de retornos de los métodos.

```

public class Animal {
    int weight, height;
    double speed;
    Animal() {
        weight = 50;
        height = 4;
        speed = 2; //miles per hour
    }
    Animal(int w, int h, int s ) {
        weight = w;
        h = height;
        speed = s
    }
    public double getTime(double miles) { //gets the number of hours to go these
miles
        return miles/speed;
    }
    public int getWeight() {
        return weight;
    }
    public int getHeight() {
        return height;
    }
    public double getSpeed() {
        return speed;
    }
}

```

3. Escriba el código para crear dos instancias de la plantilla de clase Animal incluida en el problema número 2. Asegúrese de utilizar ambos constructores proporcionados. Luego, agregue un código Java que imprima lo siguiente:
- El animal 1 tiene una velocidad de ____.
 - El animal 2 tiene una velocidad de ____.

Asegúrese de que los espacios en blanco se completen automáticamente con las velocidades reales. Utilice los métodos proporcionados para acceder a las velocidades.

4. Escriba una clase Estudiante. Debe tener las siguientes variables de instancia para el nombre, los créditos, el promedio ponderado (GPA) y los puntos de calidad. Cree un método constructor. Cree los otros dos métodos de la siguiente manera:
- Un método que devolverá el promedio ponderado actual, que serán los puntos de calidad divididos por los créditos.
 - Un método que tomará los créditos de la clase o el semestre junto con los puntos de calidad. Debe actualizar los créditos, los puntos de calidad y el GPA.
5. Con la clase creada en el número 4, cree tres instancias de la clase Estudiante de la tabla a continuación:

Nombre	Créditos	Puntos de calidad
Mary Jones	14	46
John Stiner	60	173
Ari Samala	31	69

6. Con las variables de instancias creadas en el punto 5, agregue 13 créditos y 52 puntos de calidad al estudiante "Ari Samala".
7. Utilice la clase Carta de las diapositivas y pruebe el programa para asegurarse de que funciona. Agregue una segunda Carta aleatoria. El código se incluye a continuación:

```
public class Card{
    String suit,name;
    int points;

    Card(int n1, int n2){
        suit = getSuit(n1);
        name = getName(n2);
        points = getPoints(name);
    }

    public String toString(){
        return "The " + name + " of " + suit;
    }

    public String getName(int i){
        if(i == 1) return "Ace";
    }
}
```

```

        if(i == 2)    return "Two";
        if(i == 3)    return "Three";
        if(i == 4)    return "Four";
        if(i == 5)    return "Five";
        if(i == 6)    return "Six";
        if(i == 7)    return "Seven";
        if(i == 8)    return "Eight";
        if(i == 9)    return "Nine";
        if(i == 10)   return "Ten";
        if(i == 11)   return "Jack";
        if(i == 12)   return "Queen";
        if(i == 13)   return "King";
        return "error";
    }

    public int getPoints(String n){
        if(n == "Jack" || n == "Queen" || n == "King" || n == "Ten")
            return 10;
        if(n == "Two")
            return 2;
        if(n == "Three")
            return 3;
        if(n == "Four")
            return 4;
        if(n == "Five")
            return 5;
        if(n == "Six")
            return 6;
        if(n == "Seven")
            return 7;
        if(n == "Eight")
            return 8;
        if(n == "Nine")
            return 9;
        if(n == "Ace")
            return 1;
        return -1;
    }
}

```

```

        public String getSuit(int i){
            if(i == 1)    return "Diamonds";
            if(i == 2)    return "Clubs";
            if(i == 3)    return "Spades";
            if(i == 4)    return "Hearts";
            return "error";
        }
    }

    public class Main {

        public static void main(String args[]){

            int suitNumber = (int) (Math.random()*4.0+1);
            int faceNumber = (int) (Math.random()*13.0+1);
            Card newCard = new Card(suitNumber,faceNumber);
            System.out.println(newCard);

        }

    }

```

8. Agregue un código a la clase Principal en el ejercicio 7 para hacer lo siguiente:
 - a. Mostrar el valor en puntos total de las dos cartas aleatorias.
 - b. Preguntar al usuario si quisiera otra carta. Si dicen Sí, mostrar la nueva carta y los puntos para las 3 cartas en su “Mano”.
 - c. Aplique un ciclo para permitir al usuario continuar agregando cartas a la mano hasta que la cantidad de puntos llegue a 21 o el usuario decida no agregar más cartas o el total de las cartas sea 5.