



UNIVERSIDAD NACIONAL DE INGENIERIA

Área del conocimiento de tecnología de información y
comunicación

Introducción a la programación

Docente. Ing. Cristopher Larios

Estudiante. Ronald Ramon Oporta Sequeira

Grupo. 1M7-S

Guía práctica #6

```
J App.java 2  J Empleado.java  J EmpleadoTiempoCompleto.java  J EmpleadoMedioTiempo.java X  ...
src > J EmpleadoMedioTiempo.java
1  public class EmpleadoMedioTiempo extends Empleado {
2
3      private double salarioPorHora;
4      private int horasTrabajadasPorSemana;
5
6      public EmpleadoMedioTiempo(String nombre, String numeroIdentificacion, double salarioPorHora, int horasTrabajadasPorSemana) {
7          super(nombre, numeroIdentificacion);
8          this.salarioPorHora = salarioPorHora;
9          this.horasTrabajadasPorSemana = horasTrabajadasPorSemana;
10     }
11     @Override
12     public double calcularSalario(){
13         return salarioPorHora * horasTrabajadasPorSemana * 4; //salario mensual
14     }
15
16
17 }
```

```
J App.java 2  J Empleado.java  J EmpleadoTiempoCompleto.java X  J EmpleadoMedioTiempo.java
src > J EmpleadoTiempoCompleto.java > ...
1  public class EmpleadoTiempoCompleto extends Empleado{
2
3      private double salarioAnual;
4
5      public EmpleadoTiempoCompleto(String nombre, String numeroIdentificacion, double salarioAnual){
6          super(nombre, numeroIdentificacion);
7          this.salarioAnual = salarioAnual;
8      }
9      @Override
10     public double calcularSalario() {
11         return salarioAnual / 12; //salario mensual
12     }
13
14
15
16 }
```

```
J App.java 2  J Empleado.java X  J EmpleadoTiempoCompleto.java  J EmpleadoMedioTiempo.java
src > J Empleado.java > ...
1  abstract class Empleado {
2      private String nombre;
3      private String numeroIdentificacion;
4
5      public Empleado(String nombre, String numeroIdentificacion){
6          this.nombre = nombre;
7          this.numeroIdentificacion = numeroIdentificacion;
8      }
9      public String getNombre(){
10         return nombre;
11     }
12     public String getNumeroIdentificacion(){
13         return numeroIdentificacion;
14     }
15     //metodo abstracto
16     public abstract double calcularSalario();
17 }
```

```
J App.java 2 X  J Empleado.java  J EmpleadoTiempoCompleto.java  J EmpleadoMedioTiempo.java
src > J App.java > ...
1  public class App {
2      Run | Debug
3      public static void main(String[] args) throws Exception {
4          Empleado[] empleados = new Empleado[2];
5          empleados[0] = new EmpleadoMedioTiempo(nombre:"Juan Perez", numeroIdentificacion:null, salarioPorHora:0,
6          empleados[1] = new EmpleadoMedioTiempo(nombre:"Maria Lopez", numeroIdentificacion:null, salarioPorHora:0,
7
8          for (Empleado empleado : empleados){
9              System.out.println("Empleado: " + empleado.getNombre() + ", Salario mensual: $" + empleado.calcularSa
10
11      }
12  }
13 }
```

- Responder

¿En qué parte del código usamos polimorfismo, explique?

El polimorfismo se encuentra en el código desde el modelo a seguir del empleado ya que empleado es la clase base que hereda a la clase empleado como empleado de tiempo completo o de medio tiempo, el empleado es la misma persona, pero lo que cambia es la clase del trabajo o del tiempo del trabajo que hace y es por eso que en el código se encuentran variaciones en el sueldo

¿En qué parte del código usamos abstracción, explique?

La abstracción en el código lo podemos encontrar ligado a la herencia, lo que quiero dar a entender es que la clase Empleado se abstracta para dar a la clase base y le da una estructura común a la herencia.

¿En qué parte del código usamos encapsulamiento, explique?

El encapsulamiento en el código se utiliza cuando en la clase empleado ya que el nombre y el número de identificación están en privado ya que es una información privada que no cualquiera puede obtener hasta lo encontramos en getnombre y getnumeroidentificacion.

¿En qué parte del código usamos herencia, explique?

La herencia en el código lo usamos cuando hay variaciones de empleado con empleado de tiempo completo y con empleado de medio tiempo ya que con la palabra extends y la clase base en empleado, pienso que ahí se encuentra dicha herencia.