

```
package empresainformatica;

/**
 *
 * @author Admin R30
 */

public class EmpresaInformatica {

    /**
     * @param args the command line arguments
     */

    public static void main(String[] args) {

        Empresa EsguevaCompany;

        EsguevaCompany = new Empresa(3);

        EsguevaCompany.pedirDatosEmpleados();

        EsguevaCompany.informeSalario();

    }

}
```

```
package empresainformatica;

import utilidades.Caracteres;
import utilidades.Numeros;

/**
 * @author Admin R30
 */
public class Empresa {

    Empleado[] empleados;
    Categoria[] categorias;

    /**
     * Constructor con Numero fijo de empleados.
     */
    public Empresa() {
        empleados = new Empleado[10];
        inicializar();
        crearCategorias(); }

    /**
     * Constructor con numero variable de empleados.
     *
     * @param nEmpleado Numero de empleados de la empresa.
     */
    public Empresa(int nEmpleado) {
        empleados = new Empleado[nEmpleado];
        inicializar();
        crearCategorias(); }

    /**
     * Inicializa todas las variables de empleados.
     */
    private void inicializar() {
        for (int nEmpleado = 0; nEmpleado < empleados.length; nEmpleado++) {
            empleados[nEmpleado] = new Empleado(); }
    }
}
```

```

/**
 * Crea las cuatro categorias de la empresa.
 */

private void crearCategorias() {

    categorias = new Categoria[4];

    categorias[0] = new Categoria("Administrativo", 200, 20);

    categorias[1] = new Categoria("Programador", 220, 30);

    categorias[2] = new Categoria("Analista", 250, 40);

    categorias[3] = new Categoria("Analis-Programa", 300, 50);  }

/**

 * Visualiza las propiedades de las categorias.
 */

private void informacionCategoria() {

    System.out.println("\t\tINFORME DE LAS CATEGORIAS");

    System.out.println("-----");

    System.out.println("CATEGORIA \t\t\tTARIFA \t\tEXTRAS");

    System.out.println("-----");

    for (int pos = 0; pos < categorias.length; pos++) {

        System.out.println(categorias[pos].getTarifa() + "\t\t\t" + categorias[pos].getsBase() + "\t\t" + categorias[pos].getExtras());

    }

    System.out.println("-----");  }

/**

 * Metodo que busca la existencia de la Categoria.
 *
 * @param categoriaBuscar Categoria que se pretende comparar.
 * @return pos Posicion donde se encuentra la categoria (-1 en el caso de que no la encuentre).
 */

private int busquedaCategoria(String categoriaBuscar) {

    boolean encontrado = false;

    int pos = 0;

    while (!encontrado && pos < categorias.length) {

        if (categorias[pos].getTarifa().equalsIgnoreCase(categoriaBuscar)) {

            encontrado = true;

        } else {

            pos++;

        }

    }

}

```

```
if (!encontrado) {

    System.out.println("NO SE ENCUENTRA LA CATEGORIA");

    pos = -1;

}

return pos;

}

/**

 * Pide el nombre de la categoria.

 */

public void pedirDatosEmpleados() {

    String nombre;

    int horas;

    String nombreCategoria;

    int categoria;

    Caracteres caracteres = new Caracteres();

    Numeros numeros = new Numeros();

    informacionCategoria();

    for (int nEmpleado = 0; nEmpleado < empleados.length; nEmpleado++) {

        nombre = caracteres.pedirString("NOMBRE:");

        horas = numeros.pedirnumeroEntero("HORAS: ", 0);

        nombreCategoria = caracteres.pedirString("NOMBRE CATEGORIA:");

        categoria = busquedaCategoria(nombreCategoria);//CATEGORIA

        while (categoria == -1) {

            System.out.println("CATEGORIA INEXISTENTE");

            nombreCategoria = caracteres.pedirString("NOMBRE CATEGORIA");

            categoria = busquedaCategoria(nombreCategoria);//CATEGORIA

        }

        empleados[nEmpleado].setNombre(nombre);

        empleados[nEmpleado].setHoras(horas);

        empleados[nEmpleado].setCategoria(categoria);

        System.out.println("-----");

    }

}
```

```
/**
 * Informe del salario de empleados.
 */
public void informeSalario() {
    System.out.println("-----INFORME-----");
    System.out.println("CATEGORIA: \tNOMBRE: \tHORAS EXTRA: \tSALARIO:");
    System.out.println("-----");
    for (int pos = 0; pos < empleados.length; pos++) {
        int posCategoriaEmpleado = empleados[pos].getCategoria();
        float salario = empleados[pos].Salario(
            categorias[posCategoriaEmpleado].getExtras(),
            categorias[posCategoriaEmpleado].getBase());

        System.out.println(categorias[posCategoriaEmpleado].getTarifa() + "\t" + empleados[pos].getNombre() + "\t\t" + empleados[pos].getHoras() +
            "\t\t" + salario + "€");
    }
    System.out.println("-----");
}
}
```

```
package empresainformatica;

/**
 *
 * @author Admin R30
 */
public class Empleado {

    String nombre;

    int horas;

    int categoria;

    /**
     * Constructor de empleado.
     */
    public Empleado() {

    }

    /**
     *
     * @return
     */
    public String getNombre() {

        return nombre;

    }

    /**
     *
     * @param nombre
     */
    public void setNombre(String nombre) {

        this.nombre = nombre;

    }

    /**
     *
     * @return
     */
    public int getHoras() {

        return horas;

    }

}
```

```
/**
 *
 * @param horas
 */
public void setHoras(int horas) {
    this.horas = horas;
}

/**
 *
 * @return
 */
public int getCategoria() {
    return categoria;
}

/**
 *
 * @param categoria
 */
public void setCategoria(int categoria) {
    this.categoria = categoria; }

/**
 * Calculo de salario empleado.
 *
 * @param extras Importe de la hora extra.
 * @param sBase Salario Base.
 * @return salario Salario calculado.
 */
public float Salario(float extras, float sBase) {
    float salario = 0;
    salario = horas * extras + sBase; // extras y sBase lo paso desde empresa
    return salario;
}
}
```

```
package empresainformatica;

public class Categoria {

    private String tarifa;

    private float sBase;

    private float extras;

    /**
     * @param tarifa Nombre de la tarifa.
     * @param sBase Salario Base.
     * @param extras Horas Extras.
     */
    public Categoria(String tarifa, float sBase, float extras) {

        this.tarifa = tarifa;

        this.sBase = sBase;

        this.extras = extras;  }

    /**
     *
     * @return tarifa
     */
    public String getTarifa() {

        return tarifa;  }

    /**
     *
     * @return sBase
     */
    public float getsBase() {

        return sBase;  }

    /**
     *
     * @return extras
     */
    public float getExtras() {

        return extras;

    }

}
```