



Planificación y realización de pruebas

Pruebas funcionales: caja negra

Esta práctica ha sido desarrollada en su totalidad por Román Ginés Martínez Ferrández (rgmf@riseup.net) salvo referencias al pie de página.



Todas las imágenes y todos los recursos utilizadas son de Dominio Público a menos que se diga lo contrario.

Creative Commons Reconocimiento – NoComercial - CompartirlIgual
CC by-nc-sa

Especificaciones

Hay que desarrollar un programa en el que hay un método llamado **categoría** que recibe 3 parámetros que son:

- **codigoEmpleado** es un código alfanumérico que codifica el nombre del empleado. En estos códigos alfanuméricos tienen que constar dígitos y letras minúsculas.
- **mesesTrabajados** es un número entre 0 y 999.
- **directivo** puede ser “+” o “-” para indicar que, dicho empleado, es o no un directivo, respectivamente.

El método devolverá una de las siguientes constantes, en función de los parámetros de entrada:

- **D1** para los empleados directivos con, al menos, 12 meses de antigüedad.
- **E1** para los empleados NO directivos con, al menos, 12 meses de antigüedad.
- **D2** para los empleados directivos que no lleguen a los 12 meses de antigüedad.
- **E2** para los empleados NO directivos que no lleguen a los 12 meses de antigüedad.
- **ERROR** si alguno de los parámetros no cumple las especificaciones

Entrega

Crea en tu cuenta de GitHub un repositorio llamado **eed-tema3-p5** donde subirás este documento completado y el programa desarrollado en el último punto.



Clases de equivalencia

Completa las siguientes tablas con las clases de equivalencia para cada uno de los parámetros de entrada y para la salida:

codigoEmpleado	
Código	Clase de equivalencia
E1a	Cadena con digito y minúsculas
E1b	Cadena solo con digito
E1c	Cadena solo con minúsculas
E1d	Cadena solo con mayúsculas
E1e	Cadena con digito y con mayúsculas
E1f	Cadena con mayúsculas y minúsculas y sin digito
E1g	Cadena con mayúsculas y minúsculas y digito
E1h	Cadena vacío
E1i	Cadena nula

mesesTrabajados	
Código	Clase de equivalencia
E2a	Números menores que 0
E2b	Números entre 0 y 999
E2c	Números mayores que 999
E2d	Null

directivo	
Código	Clase de equivalencia
E3a	Cadena “+”
E3b	Cadena “-”
E3c	Cadena con 2 caracteres
E3d	Cadena nula
E3e	Cadena vacía

salida



Código	Clase de equivalencia
Sa	D1
Sb	E1
Sc	D2
Sd	E2
Se	ERROR



Identifica los valores límites

Completa las siguientes tablas con los valores límites para las entradas:

codigoEmpleado	
Código	Valor límite

mesesTrabajados	
Código	Valor límite
La	Numero igual a -1
Lb	Numero igual a 0
Lc	Numero igual a 999
Ld	Numero igual a 1000

directivo	
Código	Valor límite



Casos de prueba

Completa las siguientes tablas con los casos de prueba planificados:

Clases de equivalencia					
codigoEmpleado		mesesTrabajados	directivo	salida	Códigos
"abc123"		25	+	D1	E1a, E2b, E3a, Sa
"12345"		-5	-	ERROR	E1b, E2a, E3b, Se
"abcdef"		24	+	ERROR	E1c, E2b, E3a, Se
"ABCDF"		1009	null	ERROR	E1d, E2c, E3d, Se
"ABC123"		null	""	ERROR	E1e, E2d, E3e, Se
"ABCdefg"		-5	+	ERROR	E1f, E2a, E3a, Se
"ABcd123"		30	-	ERROR	E1g, E2b, E3b, Se
null		1100	+	ERROR	E1h, E2c, E3a, Se
""		21	-	ERROR	E1i, E2b, E3b, Se
"abc123"		13	-	E1	E1a, E2b, E3b, Sb
"abc123"		9	+	D2	E1a, E2b, E3a, Sc
"abc123"		8	-	E2	E1a, E2b, E3b, Sd

Valores límite					
codigoEmpleado		mesesTrabajados	directivo	salida	Códigos
"abc123"		-1	+	ERROR	La, Se
"abc123"		0	-	E1	Lb, Sb
"abc123"		999	+	D1	Lc, Sa
"abc123"		1000	-	ERROR	Ld, Se



Román Ginés Martínez Ferrández

1º DAW: Entornos de Desarrollo



Desarrollo del programa

Por último, desarrolla un programa en Java, con una única clase llamada **Main** que contenga el método **main** y el método **categoría** y que realice lo indicado en las especificaciones.

Desde el método **main** se llamará al método **categoría**.

Crea las constantes siguientes en este programa para devolver los resultados oportunos:

```
private static final int ERROR = 0;  
private static final int D1 = 11;  
private static final int D2 = 12;  
private static final int E1 = 21;  
private static final int E2 = 22;
```

Así, la única clase de este programa tiene la siguiente estructura:

```
public class Main {  
  
    private static final int ERROR = 0;  
    private static final int D1 = 11;  
    private static final int D2 = 12;  
    private static final int E1 = 21;  
    private static final int E2 = 22;  
  
    public static void main(String[] args) {  
        Main programa = new Main();  
        int resultado = programa.categoría("fds394", 32, "+");  
    }  
  
    public int categoría(  
        String codigoEmpleado,  
        int mesesTrabajado,  
        String directivo) {  
        // TODO Elimina este return y completa este método  
        return 0;  
    }  
}
```