

Particiones de equivalencias y de valores limites

Estudiante:

Método setMatricula(String matricula);

Asignar matricula si: "[0-2][0-9]00[1-5]([0-9]{4})"

La expresión regular [0-2][0-9]00[1-5]([0-9]{4}) se desglosa de la siguiente manera:

[0-2]: Este grupo indica que el primer dígito puede ser 0, 1 o 2.

[0-9]: Después del primer dígito, puede haber cualquier dígito del 0 al 9.

00: Estos dos ceros deben aparecer juntos después del primer dígito.

[1-5]: Después de los ceros, el siguiente dígito debe estar entre 1 y 5.

([0-9]{4}): Este grupo indica que debe haber exactamente 4 dígitos del 0 al 9 después del dígito entre 1 y 5.

En resumen, esta expresión regular coincide con cualquier secuencia de caracteres que comience con un número entre 0 y 2, seguido de cualquier número, seguido de "00", seguido de un número entre 1 y 5, y terminando con exactamente 4 dígitos del 0 al 9.

Los primeros dos dígitos corresponden con el año, dos ceros, el número de plantel (casa libertad, Cuauhtepc, san lorenzo tezonco, del valle y centro histórico), por último, el número de estudiante conformado por 4 dígitos.

Rangos Naturales

Rango de la Expresión Regular [0-2][0-9]00[1-5]([0-9]{4})

Rango de Estado Matricula_valida Matricula_No_Valida

Rangos Basados en la Especificación

Entradas

Matricula	[0-2][0-9]00[1-5]([0-9]{4})
-----------	-----------------------------

Salida

Estado	[Matricula valida], [Matricula No Valida]
--------	-------------------------------------------

Particiones de Equivalencia de Entrada

TCI Test Coverage Item (Elemto de cobertura de prueba)

PEE partición de Equivalencia de Entrada

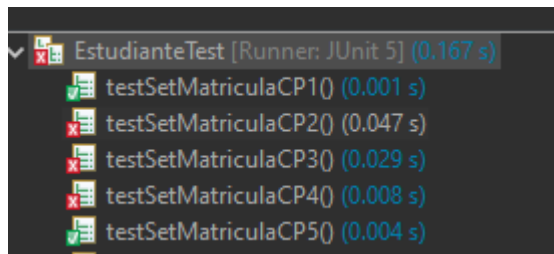
PES partición de Equivalencia de Salida

TABLA DE PARTICIONES DE EQUIVALENCIA			
TCI	Unidad Bajo Prueba	Parámetro	Rangos
PEE1 *	setMatricula()	Matricula	
PEE2			[0-2][0-9]00[1-5]([0-9]{4})
PEE3		Estado	[Matricula Valida]
PEE4			[Matricula No Valida]

TABLA DE CASOS DE PRUEBA (TC Test Case) PARA PARTICIONES DE EQUIVALENCIA			
ID	Particiones de Equivalencia	Entrada	Salida Esperada
CP1	PEE2, PEE3	180030773	Matricula Valida
CP2	[PEE2], PEE4	022001234	Matricula No Valida
CP3	[PEE2], PEE4	Hola mundo	Matricula No Valida
CP4	[PEE2], PEE4	9	Matricula No Valida
CP5	[PEE2], [PEE3]	100028437	Matricula Valida

TABLA DE VALORES LÍMITE			
ID	Unidad Bajo Prueba	Parámetro	Valor Limite
	setMatricula()	Matricula	
VL1			[0-2][0-9]00[1-5]([0-9]{4})
VL2		Estado	
VL3			Matricula Valida
VL4			Matricula No Valida

TABLA DE CASOS DE PRUEBA DE ANÁLISIS DE VALORES LÍMITE			
ID	Valores Limite	Entrada	Salida Esperada
CP1	VL1, VL3	180030773	Matricula Valida
CP2*	[VL1], VL4	022001234	Matricula No Valida
CP3*	[VL1], [VL4]	Hola mundo	Matricula No Valida
CP4*	[VL1], [VL4]	9	Matricula No Valida
CP5	[VL1], [VL3]	100028437	Matricula Valida



```
@Test
void testSetMatriculaCP1() throws Exception{
    estudiante.setMatricula("180030773");
}

@Test
void testSetMatriculaCP2() throws Exception{
    estudiante.setMatricula("022001234");
}

@Test
void testSetMatriculaCP3() throws Exception{
    estudiante.setMatricula("Hola mundo");
}

@Test
void testSetMatriculaCP4() throws Exception{
    estudiante.setMatricula("9");
}

@Test
void testSetMatriculaCP5() throws Exception{
    estudiante.setMatricula("100028437");
}
```

Medico:

Método setCedulaProfesional()

Asignar Cedula Profesional si: “([A-Z]{4})([0-9]{6})(H|M)([A-Z]{6})([0-9]{2})([0-9]{10})”

Investigando, se nos indica que un número de cédula profesional esta conformado por la CURP de la persona seguido de una serie de números que pueden indicar la institución emisora, el estado o la especialidad del médico, para nuestro caso, supondremos un margen de 10 números.

La expresión regular ([A-Z]{4})([0-9]{6})(H|M)([A-Z]{6})([0-9]{2})([0-9]{10}) se desglosa de la siguiente manera:

([A-Z]{4}): Este grupo captura cuatro letras mayúsculas consecutivas, que generalmente representan los apellidos de la persona.

([0-9]{6}): Este grupo captura seis dígitos numéricos consecutivos, que comúnmente representan la fecha de nacimiento en el formato año/mes/día.

(H|M): Este grupo captura la letra "H" o "M", que suele indicar el género de la persona (H para hombre, M para mujer).

([A-Z]{6}): Este grupo captura seis letras mayúsculas consecutivas, que usualmente representan el primer nombre, la inicial del segundo nombre y el primer apellido.

([0-9]{2}): Este grupo captura dos dígitos numéricos consecutivos, que a menudo representan la entidad federativa de nacimiento.

([0-9]{10}): Este grupo captura diez dígitos numéricos consecutivos, que, para nuestro caso, lo tomaremos como los datos de la institución emisora, estado o especialidad del médico.

Rangos Naturales

Rango de la Expresión Regular:([A-Z]{4})([0-9]{6})(H|M)([A-Z]{5})([0-9]{2})([0-9]{10})

Rango de Estado Cedula_Profesional_Valida Cedula_Profesional_No_Valida

Rangos Basados en la Especificación

Entradas

Cedula Profesional	([A-Z]{4})([0-9]{6})(H M)([A-Z]{5})([0-9]{2})([0-9]{10})
--------------------	----------------------------------------------------------

Salida

Estado	[Cedula_Profesional_Valida], [Cedula_Profesional_No_Valida]
--------	-------------------------------------------------------------

Particiones de Equivalencia de Entrada

TCI Test Coverage Item (Elemto de cobertura de prueba)

PEE partición de Equivalencia de Entrada

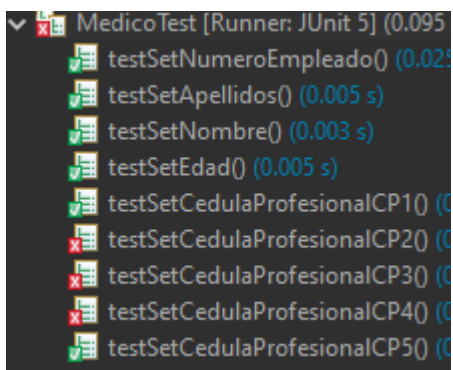
PES partición de Equivalencia de Salida

TABLA DE PARTICIONES DE EQUIVALENCIA			
TCI	Unidad Bajo Prueba	Parámetro	Rangos
PEE1 *	setCedulaProfesional()	cedulaProfesional	
PEE2			([A-Z]{4})([0-9]{6})(H M)([A-Z]{5}) ([0-9]{2})([0-9]{10})
PEE3		Estado	[Cedula Profesional Valida]
PEE4			[Cedula Profesional No Valida]

TABLA DE CASOS DE PRUEBA (TC Test Case) PARA PARTICIONES DE EQUIVALENCIA			
ID	Particiones de Equivalencia	Entrada	Salida Esperada
CP1	PEE2, PEE3	JUCA800215HDFLRN091234567890	Cedula Profesional Valida
CP2	[PEE2], PEE4	7813861084723890	Cedula_Profesional_No_Valida
CP3	[PEE2], PEE4	Hola mundo	Cedula_Profesional_No_Valida
CP4	[PEE2], PEE4	9	Cedula_Profesional_No_Valida
CP5	[PEE2], [PEE3]	CUFK990810HDFLLV045527529715	Cedula_Profesional_Valida

TABLA DE VALORES LÍMITE			
ID	Unidad Bajo Prueba	Parámetro	Valor Limite
	setCedulaProfesional()	cedulaProfesional	
VL1			([A-Z]{4})([0-9]{6})(H M)([A-Z]{5}) ([0-9]{2})([0-9]{10})
VL2		Estado	
VL3			Cedula Profesional Valida
VL4			Cedula Profesional No Valida

TABLA DE CASOS DE PRUEBA DE ANÁLISIS DE VALORES LÍMITE			
ID	Valores Limite	Entrada	Salida Esperada
CP1	VL1, VL3	JUCA800215HDFLRN091234567890	Cedula_Profesional_Valida
CP2*	[VL1], VL4	7813861084723890	Cedula_Profesional_No_Valida
CP3*	[VL1], [VL4]	Hola mundo	Cedula_Profesional_No_Valida
CP4*	[VL1], [VL4]	9	Cedula_Profesional_No_Valida
CP5	[VL1], [VL3]	CUFK990810HDFLLV045527529715	Cedula_Profesional_Valida



```

@Test
void testSetCedulaProfesionalCP1() throws Exception{
    medico.setCedulaProfesional("JUCA800215HDFLRN091234567890");
}

@Test
void testSetCedulaProfesionalCP2() throws Exception{
    medico.setCedulaProfesional("7813861084723890");
}

@Test
void testSetCedulaProfesionalCP3() throws Exception{
    medico.setCedulaProfesional("Hola mundo");
}

@Test
void testSetCedulaProfesionalCP4() throws Exception{
    medico.setCedulaProfesional("9");
}

@Test
void testSetCedulaProfesionalCP5() throws Exception{
    medico.setCedulaProfesional("CUFK990810HDFLLV045527529715");
}

```

Método setNumeroEmpleado();

Asignar Número de empleado si: [0-9]{5}

Para este caso, decidimos que nuestro número de empleado esta conformado por 5 números consecutivos.

Rangos Naturales

Rango de la Expresión Regular: [0-9]{5}

Rango de Estado Numero_Empleado_Valido Numero_Empleado_No_Valido

Rangos Basados en la Especificación

Entradas

NumeroEmpleado	[0-9]{5}
----------------	----------

Salida

Estado	[Numero_Empleado_Valido], [Numero_Empleado_No_Valido]
--------	-------------------------------------------------------

Particiones de Equivalencia de Entrada

TCI Test Coverage Item (Elemto de cobertura de prueba)

PEE partición de Equivalencia de Entrada

PES partición de Equivalencia de Salida

TABLA DE PARTICIONES DE EQUIVALENCIA			
TCI	Unidad Bajo Prueba	Parámetro	Rangos
PEE1 *	setNumeroEmpleado()	numeroEmpleado	
PEE2			[0-9]{6}
PEE3		Estado	[Numero_Empleado_Valido]
PEE4			[Numero_Empleado_No_Valido]

TABLA DE CASOS DE PRUEBA (TC Test Case) PARA PARTICIONES DE EQUIVALENCIA			
ID	Particiones de Equivalencia	Entrada	Salida Esperada
CP1	PEE2, PEE3	12345	Numero_Empleado_Valido
CP2	[PEE2], PEE4	7813861084723890	Numero_Empleado_No_Valido
CP3	[PEE2], PEE4	Hola mundo	Numero_Empleado_No_Valido
CP4	[PEE2], PEE4	9	Numero_Empleado_No_Valido

CP5	[PEE2], [PEE3]	43654	Numero_Empleado_Valido
-----	-------------------	-------	------------------------

TABLA DE VALORES LÍMITE			
ID	Unidad Bajo Prueba	Parámetro	Valor Limite
	setNumeroEmpleado()	numeroEmpleado	
VL1			[0-9]{5}
VL2		Estado	
VL3			Numero_Empleado_Valido
VL4			Numero_Empleado_No_Valido

TABLA DE CASOS DE PRUEBA DE ANÁLISIS DE VALORES LÍMITE			
ID	Valores Limite	Entrada	Salida Esperada
CP1	VL1, VL3	12345	Numero_Empleado_Valido
CP2*	[VL1], VL4	7813861084723890	Numero_Empleado_No_Valido
CP3*	[VL1], [VL4]	Hola mundo	Numero_Empleado_No_Valido
CP4*	[VL1], [VL4]	9	Numero_Empleado_No_Valido
CP5	[VL1], [VL3]	46354	Numero_Empleado_Valido

```

@Test
void testSetNumeroEmpleadoCP1() throws Exception{
    medico.setNumeroEmpleado("12345");
}

@Test
void testSetNumeroEmpleadoCP2() throws Exception{
    medico.setNumeroEmpleado("7813861084723890");
}

@Test
void testSetNumeroEmpleadoCP3() throws Exception{
    medico.setNumeroEmpleado("Hola mundo");
}

@Test
void testSetNumeroEmpleadoCP4() throws Exception{
    medico.setNumeroEmpleado("9");
}

@Test
void testSetNumeroEmpleadoCP5() throws Exception{
    medico.setNumeroEmpleado("43654");
}

```

MedicoTest [Runner: JUnit 5] (0.095 s)

- testSetApellidos() (0.034 s)
- testSetNumeroEmpleadoCP1() (0.002 s)
- testSetNumeroEmpleadoCP2() (0.009 s)
- testSetNumeroEmpleadoCP3() (0.004 s)
- testSetNumeroEmpleadoCP4() (0.003 s)
- testSetNumeroEmpleadoCP5() (0.002 s)
- testSetNombre() (0.001 s)
- testSetEdad() (0.001 s)
- testSetCedulaProfesionalCP1() (0.002 s)
- testSetCedulaProfesionalCP2() (0.006 s)
- testSetCedulaProfesionalCP3() (0.004 s)