Documentación de Pruebas unitarias

Tatiana Sanchez

# Prueba-Caso 1

|  |  |  |  |
| --- | --- | --- | --- |
| CASO DE PRUEBA | ENTRADA A+B | RESULTADO ESPERADO | Resultado  obtenido |
| 1. Suma de dos valores enteros positivos en variables a y b | a = 3  b = 3 | 6 | Resultado= 6 |
| 1. Las variables a y b esta vez tomaron un valor positivo para la a y uno negativo para la b | a = -3  b= 3 | 0 | Resultado = 0 |
| 1. Las variables a y b, tomaran dos valores negativos | a = -3  b= -3 | -6 | Resultado = -6. |

# Prueba-Caso 2

|  |  |  |  |
| --- | --- | --- | --- |
| CASO DE PRUEBA | ENTRADA | RESULTADO ESPERADO | RESULTADO OBTENIDO |
| 1. La variable edad vas a tener 40 como valor | Edad = 40 | resultadoEsperado= true | Resultado= TRUE. |
| 1. La variable edad toma el valor 7 | Edad = 7 | resultadoEsperado= false | Resultado= False |

# Prueba-Caso 3

|  |  |  |  |
| --- | --- | --- | --- |
| CASO DE PRUEBA | ENTRADA | RESULTADO ESPERADO | RESULTADO OBTENIDO |
| 1.Se concatenan las dos cadenas de String tomando dos valores | a = “JUNIT”  b = “5” | resultadoEsperado= “JUNIT5” | Se obtuvo la cadena “JUNIT5” al concatenar las dos variables tipo String. |
| 2. Se concatenan las dos cadenas de String tomando dos valores | a = “Hola”  b = “mundo” | resultadoEsperado= “Hola mundo” | Se obtuvo un error aserción. Porque no son iguales al concatenarse. |

# Prueba-Caso 4

|  |  |  |  |
| --- | --- | --- | --- |
| CASO DE PRUEBA | ENTRADA | RESULTADO ESPERADO | RESULTADO OBTENIDO |
| 1.Se utiliza la función para determinar si el usuario es admin o no. | Nombre= “admin” | Null | El resultado obtenido efectivamente es Null. |
| 2. Se utiliza la función para determinar si el usuario es admin o no. | Nombre=”tati” | “tati” | Se obtuvo “tati” |

# Prueba-Caso 5

|  |  |  |  |
| --- | --- | --- | --- |
| CASO DE PRUEBA | ENTRADA | RESULTADO ESPERADO | RESULTADO OBTENIDO |
| 1. Se le da un valor mayor a 100,000$ | Precio=500,000$ | PrecioFinal=450,000 | Se aplico el descuento de 10% a la compra |
| 1. Se le da un valor menor a 100,000 | Precio= 50,000$ | PrecioFinal=50,000$ | No se aplica el descuento a la compra |

# Prueba-Caso 6

|  |  |  |  |
| --- | --- | --- | --- |
| CASO DE PRUEBA | ENTRADA | RESULTADO ESPERADO | RESULTADO OBTENIDO |
| 1. Se pondrá una palabra que sea un palíndromo | Palabra = “oso” | TRUE | Resultado es TRUE |
| 1. Se agrega una palabra QUE NO ES palíndromo | Palabra= “casa” | FALSE | Resultado FALSE |
| 1. Se agrega el valor NULL a la palabra | Palabra= null | FALSE | Resultado FALSE |

# Prueba-Caso 7

|  |  |  |  |
| --- | --- | --- | --- |
| CASO DE PRUEBA | ENTRADA | RESULTADO ESPERADO | RESULTADO OBTENIDO |
| 1. se agrega una ArrayList, se le adiciona una tarea. | ArrayList con una tarea agregada “pintar” | NULL | El resultado no es nulo porque el Arraylist tiene al menos una tarea |
| 1. Se agrega un ArrayList Vacio | ArrayList vacio y se llama al método ObtenerTareas() | Que el resultado sea NULL | Se obtiene el resultado NULL porque esta vacio el ArrayList |

# Prueba-Caso 8

|  |  |  |  |
| --- | --- | --- | --- |
| CASO DE PRUEBA | ENTRADA | RESULTADO ESPERADO1 | RESULTADO OBTENIDO |
| 1. SE AGREGA UNA CONTRASEÑA FUERTE | Contraseña= “tati12345” | TRUE | TRUE |
| 1. Se agrega una contraseña corta | Contraseña= “ta” | FALSE | FALSE |
| 1. Se agraga una contraseña sin números (débil) | Contraseña=”tatiii” | FALSE | FALSE, la contraseña no contiene números y por eso es débil. |

# Prueba-Caso 9

|  |  |  |  |
| --- | --- | --- | --- |
| CASO DE PRUEBA | ENTRADA | RESULTADO ESPERADO | RESULTADO OBTENIDO |
| 1. Encontrar un usuario con correo existente y que su nombre de usuario es correcto. | User(“lucia123”, “lucia@example.com”) | Que se pueda encontrar el correo y el usuario | Se encontraron los dos elementos del User porque existen. |
| 1. Buscamos encontrar un usuario con correo inexistente al igual que el usuario | User(“lucia123”, “lucia2@example.com”), | NULL | Se obtiene null porque se cumple la condición del método finderUserbyemail |

# Prueba-Caso 10

|  |  |  |  |
| --- | --- | --- | --- |
| CASO DE PRUEBA | ENTRADA | RESULTADO DE ESPERADO | RESULTADO OBTENIDO |
| 1. Se pasa como argumento al método para verificar que es factorial el numero 5 | Numero = 5 | resultadoEsperado= 1202 | Efectivamente el sistema corre sin problemas porque la factorial de 5 es 120 |
| 1. Se pasa como argumento al método para verificar que es factorial el numero 0 | Numero = 0 | resultadoEsperado=1 | Efectivamente el sistema corre sin problemas porque la factorial de 0 es 1 |
| 1. Se pasa como argumento al método para verificar que es factorial el numero 5 pero se verificara que la factorial 2. NO ES 121 | Numero= 5 | resultadoEsperado= 121 | El test da un error de aserción porque el algoritmo detecta que el factorial no es el mismo al que se espera en la prueba. |