

22.4 Test Design Techniques: Pairwise testing: Homework

Apply the pairwise to the functionality, you cover with test cases for the previous HW.

Output: 2-3 test cases using pairwise testing.

API: User and Role.

New user creation demands 7 parameters: first name, middle name, last name, username, e-mail, password, role name.

1. Let's define the equivalence classes and boundary values for each parameter:

First name (varchar(255), string): *Only alphabets, only numbers, letters (alphabets)+numbers, numbers + special characters, only special characters, alphabets + special characters, space inside, only spaces.*

Boundaries:

minimum length, empty field (not entered value), maximum length (255), maximum length (255)+/- 1.

Middle name (varchar(255), string): *Only alphabets, only numbers, letters (alphabets)+numbers, numbers + special characters, only special characters, alphabets + special characters, space inside, only spaces.*

Boundaries:

minimum length, empty field (not entered value), maximum length (255), maximum length (255)+/- 1.

Last name (varchar(255), string): *Only alphabets, only numbers, letters (alphabets)+numbers, numbers + special characters, only special characters, alphabets + special characters, space inside, only spaces.*

Boundaries:

minimum length, empty field (not entered value), maximum length (255), maximum length (255)+/- 1.

Username (varchar(255), string): *Only alphabets, only numbers, letters (alphabets)+numbers, numbers + special characters, only special characters, alphabets + special characters, space inside, only spaces.*

Boundaries:

minimum length, empty field (not entered value), maximum length (255), maximum length (255)+/- 1.

E-mail (varchar(255), string):

Positive values:

Local-part:

- uppercase and lowercase Latin letters
- digits 0 to 9
- dot (.), underscore (_), hyphen (-). dot ., provided that it is not the first or last character and provided also that it does not appear consecutively (e.g., John..Doe@example.com is not allowed)
- The maximum total length of the local-part of an email address is 64 octets.

Domain:

- uppercase and lowercase Latin letters
- Digits 0 to 9, provided that top-level domain names are not all-numeric;
- Hyphen -, provided that it is not the first or last character.

@ sing between local part and domain.

Negative values:

Non-latin letters, restricted characters !#\$%&'*/+=?^`{|}~

Only Latin letters, only numbers, Latin letters+numbers, numbers + special characters, Latin letters+ numbers + special characters, only special characters, Latin letters + special characters, space inside, only spaces,

email address without @, email address with @

Boundaries:

minimum length, empty field (not entered value), maximum Local-part length (64), maximum Local-part length (64) +/-1, maximum length (255)+/- 1, email address without @

Password (varchar(255), string): *Only alphabets, only numbers, letters*

(alphabets)+numbers, numbers + special characters, only special characters, alphabets + special characters, space inside, only spaces, mandatory special symbol "@", special.

Boundaries:

minimum length, empty field (not entered value), maximum length (255), maximum length (255)+/- 1.

Role name (varchar(255), string): *Only alphabets, only numbers, letters*

(alphabets)+numbers, numbers + special characters, only special characters, alphabets + special characters, space inside, only spaces.

Boundaries:

minimum length, empty field (not entered value), maximum length (255), maximum length (255)+/- 1.

2. To cover all the classes, let's use the following values:

First name (varchar(255), string) "Valeria", "1234567890", "Valeria17", "12345+-%@"

,"!@#\$\$%^&*()_+ = -", "Valeria!@#\$\$%", "Va leria", " ", "A", "1", "AB", "",

"qwertyuiopasdfghjklqwertyuiopzxcvbnmbvqazxswedcvfrtgbnhuyqwertyuiopasdfghjklqwe

rtuiopzxcvbnmnbvqazxswedcvfrtgbnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujRTqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvtYUIJHGFFRTTYqazxswedcvfrtgbnhyuj" (255)

"qwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujRTqHwertyuiopasdfghjklqwertyuiopzxcvbnmnbvtYUIJHGFFRTTYqazxswedcvfrtgbnhyuj"(256)

"qwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujRTqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvtYUIJHGFFRTTYqazxswedcvfrtgbnhyuj"(254)

Middle name (varchar(255), string) "Subbordew", "1234567890", "Subbordew 17", "12345+-%@" , " !@#\$\$%^&*()_+ = - ", "Subbordew!@#\$\$%", "Subb ordew", " ", "A", "1", "AB", "" ,

"qwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujRTqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvtYUIJHGFFRTTYqazxswedcvfrtgbnhyuj" (255)

"qwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujRTqHwertyuiopasdfghjklqwertyuiopzxcvbnmnbvtYUIJHGFFRTTYqazxswedcvfrtgbnhyuj"(256)

"qwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujRTqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvtYUIJHGFFRTTYqazxswedcvfrtgbnhyuj"(254)

Last name (varchar(255), string) "Anderson", "1234567890", "Anderson 17", "12345+-%@" , " !@#\$\$%^&*()_+ = - ", "Anderson!@#\$\$%", "Ander son", " ", "A", "1", "Ab", "" ,

"qwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujRTqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvtYUIJHGFFRTTYqazxswedcvfrtgbnhyuj" (255)

"qwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujRTqHwertyuiopasdfghjklqwertyuiopzxcvbnmnbvtYUIJHGFFRTTYqazxswedcvfrtgbnhyuj"(256)

"qwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujRTqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvtYUIJHGFFRTTYqazxswedcvfrtgbnhyuj"(254)

Username (varchar(255), string) "clare", "1234567890", "clare17", "12345+-%@" , " !@#\$\$%^&*()_+ = - ", "clare!@#\$\$%", "cla re", " ", "A", "1", "AB", "" ,

"qwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmnbvqazxswedcvfrtgbnhyuj"

"qwertyuiopasdfghjklqwertyuiopzxcvbnmbvqazxswedcvfrtgbnhuyjqwertyuiopasdfghjklqwertyuiopzxcvbnmbvqazxswedcvfrtnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmbvqazxswedcvfrtgbnhuyjRTqwertyuiopasdfghjklqwertyuiopzxcvbnmbvtYUIJHGFFRTTYqazxswedcvfrtgbnhuyj" (254)

[illegible]

"qwertyuiopasdfghjklqwertyuiopzxcvbnmbvqazxswedcvfrtgbnhuyjqwertyuiopasdfghjklqwertyuiopzxcvbnmbvqazxswedcvfrtnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmbvqazxswedcvfrtgbnhuyjRTqwertyuiopasdfghjklqwertyuiopzxcvbnmbvtYUIJHGFFRTTYqazxswedcvfrtgbnhuyj"(254)

Role name (varchar(255), string) "rolename""role_name", "1234567890", "rolename17", "12345+-%@" , "!@#\$\$%^&*()_+=-", "rolename!@#\$\$%", "role name", " ", "A", "1", "AB", "", "qwertyuiopasdfghjklqwertyuiopzxcvbnmbvqazxswedcvfrtgbnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmbvqazxswedcvfrtgbnhyujqwertyuiopzxcvbnmbvqazxswedcvfrtgbnhyujRTqwertyuiopasdfghjklqwertyuiopzxcvbnmbvtYUIJHGFFRTTYqazxswedcvfrtgbnhyuj" (255)

"qwertyuiopasdfghjklqwertyuiopzxcvbnmbvqazxswedcvfrtgbnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmbvqazxswedcvfrtgbnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmbvqazxswedcvfrtgbnhyujRTqHwertyuiopasdfghjklqwertyuiopzxcvbnmbvtYUIJHGFFRTTYqazxswe dcvfrtgbnhyuj"(256)

"qwertyuiopasdfghjklqwertyuiopzxcvbnmbvqazxswedcvfrtgbnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmbvqazxswedcvfrtgbnhyujqwertyuiopasdfghjklqwertyuiopzxcvbnmbvqazxswe dcvfrtgbnhyujRTqwertyuiopasdfghjklqwertyuiopzxcvbnmbvtYUIJHGFFRTTYqazxswedcvfrtgbnhyuj"(254)

- Let's not take into consideration the values that give us an error anyway (negative tests):

First name (varchar(255), string) "1234567890", "Valeria17", "12345+-%@" , "!@#\$\$%^&*()_+=-", "Valeria!@#\$\$%", " ", "1", "".

Middle name (varchar(255), string) "1234567890", "Subbordew 17", "12345+-%@" , "!@#\$\$%^&*()_+=-", "Subbordew!@#\$\$%", " ", "1", "".

Last name (varchar(255), string) "1234567890", "Anderson 17", "12345+-%@" , "!@#\$\$%^&*()_+=-", "Anderson!@#\$\$%", " ", "1", "".

Username (varchar(255), string) "!@#\$\$%^&*()_+=-", " ", "".

E-mail (varchar(255), string) test, 1234567890, 1234567890@1234567890.123, .test@test.com, test@1233, test@test, test@test.com, test@-test.com, test@test.com-,

Password (varchar(255), string) "pass word", "!@#\$\$%^&*()_+=-", " ", "A", "1", "AB", "",

Role name (varchar(255), string) "12345+-%@" , "!@#\$\$%^&*()_+=-", "rolename!@#\$\$%", " ", " ", "".

- The rest values could be used to form combinations with teremok.com.

	First name	Middle name	Last name	Username	E-mail	Password	Role
× Row 1	Valerio	Subbordew	Anderson	clare	test@test.com	password	rolename
× Row 2	Va leria	Subb ordew	Ander son	1234567890	test777@test.com	1234567890	role_name
× Row 3	A	A	A	clare17	test-777@test.com	password17	rolename17
× Row 4	Ab	Ab	Ab	12345+-%@"	test7.77@test.com	password!@#\$\$%	role name
× Row 5	qwertyuiopasdfghjklc	qwertyuiopasdfghjklc	qwertyuiopasdfghjklc	clare!@#\$\$%	test.777@test.com	qwertyuiopasdfghjklc	A
× Row 6	qwertyuiopasdfghjklc	qwertyuiopasdfghjklc	qwertyuiopasdfghjklc	cla re	test@te-st.com	qwertyuiopasdfghjklc	1
× Row 7	qwertyuiopasdfghjklc	qwertyuiopasdfghjklc	qwertyuiopasdfghjklc	A	test@12345.com	qwertyuiopasdfghjklc	Ab
× Row 8	{your value}	{your value}	{your value}	1	test@test-test.com	{your value}	qwertyuiopasdfghjklc

× Row 9	{your value}	{your value}	{your value}	Ab	testtesttesttesttes	{your value}	qwertyuioposdfghjklc
× Row 10	{your value}	{your value}	{your value}	qwertyuioposdfghjklc	testtesttesttesttes	{your value}	qwertyuioposdfghjklc
× Row 11	{your value}	{your value}	{your value}	qwertyuioposdfghjklc	testtesttesttesttes	{your value}	{your value}
× Row 12	{your value}	{your value}	{your value}	qwertyuioposdfghjklc	testtesttesttesttes	{your value}	{your value}
× Row 13	{your value}	{your value}	{your value}	{your value}	testtesttesttesttes	{your value}	{your value}
× Row 14	{your value}	{your value}	{your value}	{your value}	testtesttesttesttes	{your value}	{your value}
× Row 15	{your value}	{your value}	{your value}	{your value}	{your value}	{your value}	{your value}

- Using the teremok.com, let's create a table, generate combinations. The result table attached to the HW.

Here teremok.com couldn't handle the volume to generate All combinations. Thus, generated just Pairwise table.

- Let's choose the most interesting combinations and create test cases in TMS:

<https://app.qacoverage.com/Mentorpiece/project/M20/test-case/1/M20-TC1866>

<https://app.qacoverage.com/Mentorpiece/project/M20/test-case/1/M20-TC1867>

<https://app.qacoverage.com/Mentorpiece/project/M20/test-case/1/M20-TC1868>