1.

The password is not encrypted

app.get('/users', (req, res) => {

req.app.get('db').users.insert({

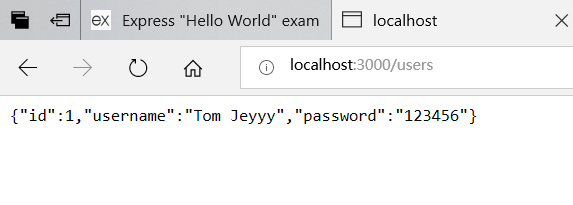
'username': "Tom Jeyyy",

'password': "123456"

}).then(users => {res.send(users);

});

});



Password encryption

Procedure:

CREATE OR REPLACE FUNCTION procedure\_title(

username1 character varying,

password1 character varying)

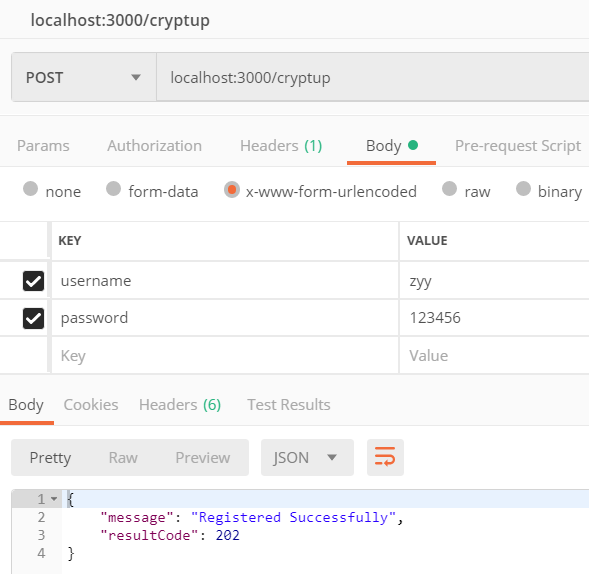
RETURNS void

LANGUAGE 'sql'

AS $BODY$

insert into users(username, password) values (username1,crypt(password1,gen\_salt('md5')))

$BODY$;



Localhost:3000/listuser



Procedure:

CREATE OR REPLACE FUNCTION cyrpyin(

username1 character varying,

password1 character varying)

RETURNS SETOF users

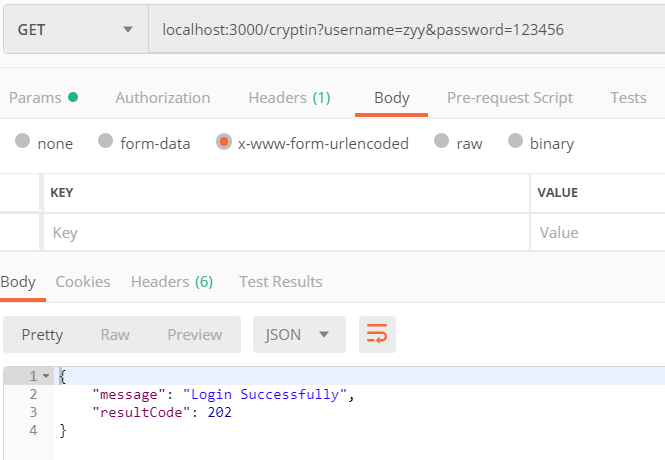
LANGUAGE 'sql'

AS $BODY$

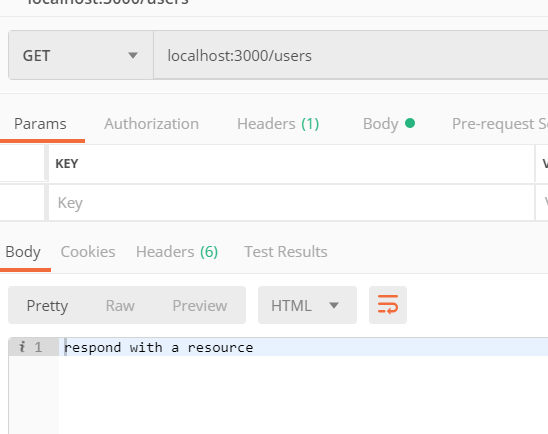
select \* from users where username =username1 and password = crypt(password1,password)

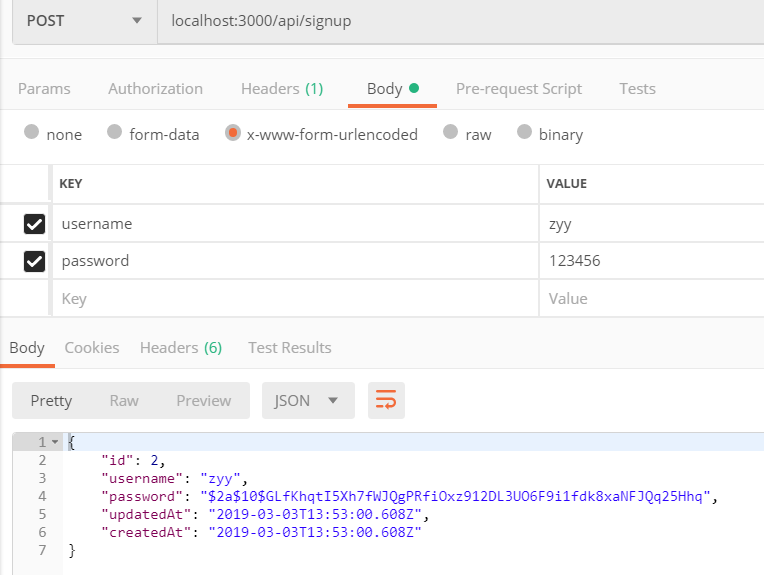
$BODY$;

User Login

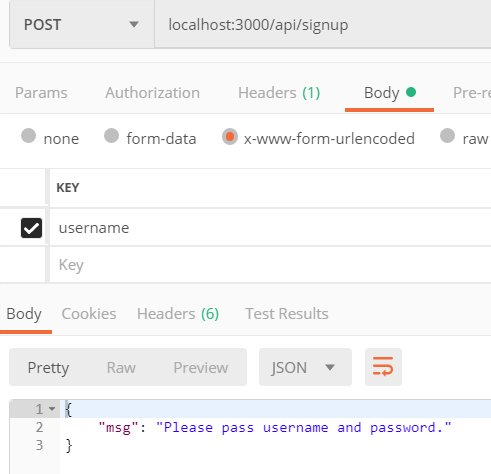


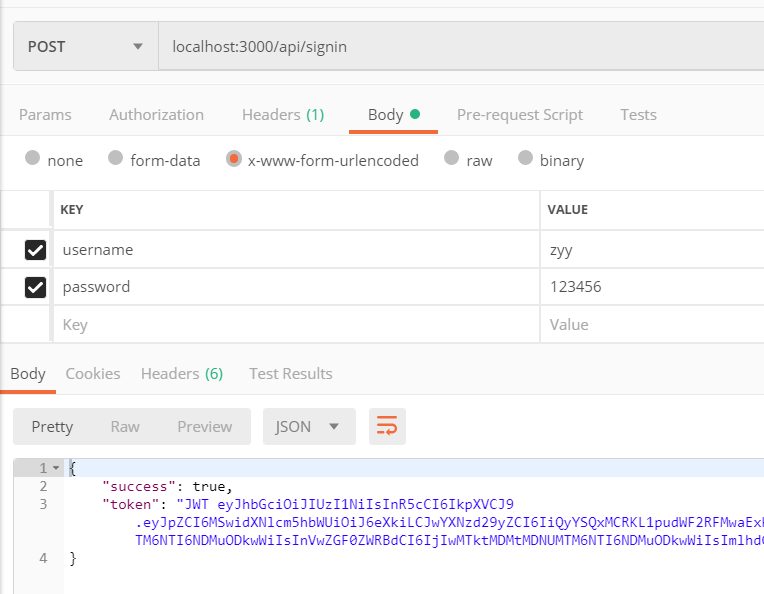
2.





If you do not add the password section

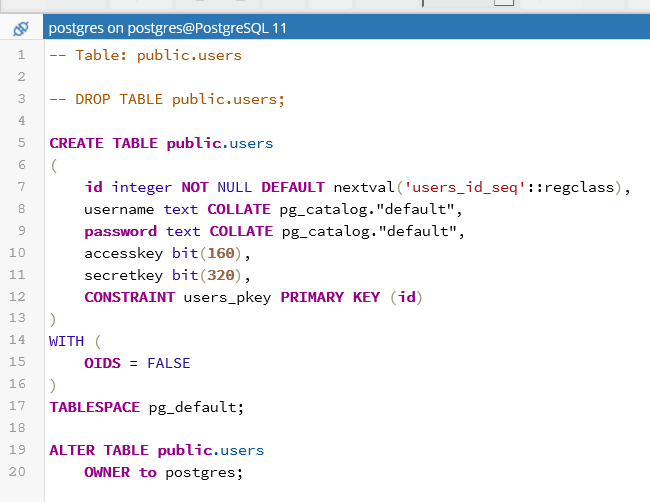




3.

alter table users add COLUMN accesskey bit(160)

alter table users add COLUMN secretkey bit(320)



4.

Procedure:

CREATE OR REPLACE FUNCTION hmac\_p\_up(

username1 character varying,

password1 character varying,

accesskey1 character varying

)

RETURNS void

LANGUAGE 'sql'

AS $BODY$

insert into users(username, password,accesskey)

values (username1,crypt(password1,gen\_salt('md5')),cast(hmac('123456',accesskey1,'md5') as text))

$BODY$;

CREATE OR REPLACE FUNCTION hmac\_p\_in(

username1 character varying,

password1 character varying,

accesskey1 character varying

)

RETURNS setof users

LANGUAGE 'sql'

AS $BODY$

select \* from users where username = username1 and password = crypt('123456',password) and accesskey =cast(hmac('123456',accesskey1,'md5') as text)

$BODY$;

