

AWS Cognito Query

Find cognito user query.

```
import AWS from 'aws-sdk'

AWS.config.update({region : config.cognito.REGION , accessKeyId :
config.cognito.ACCESS_KEY_ID,secretAccessKey :
config.cognito.SECRET_ACCESS_KEY});
var cognitoidentityserviceprovider = new AWS.CognitoIdentityServiceProvider();

const params = {
  "UserPoolId" : config.cognito.USER_POOL_ID,
  "AttributesToGet": [
    "email",
  ],
  "Filter" : `username = \`${username}\``
};

cognitoidentityserviceprovider.listUsers(params,async(err,data) =>{
  if(err){
    console.log("err",err);
  }else{
    console.log("data",data);
  }
})
```

update cognito user query.

```
const params1 = {
  UserAttributes : [
    {
      Name : "phone_number",
      Value : contactno1
    }
  ],
  UserPoolId : config.cognito.USER_POOL_ID,
  Username : username
};

cognitoidentityserviceprovider.adminUpdateUserAttributes(params1,(errr,data1) =>{
  if(errr){
    console.log("errr",errr);
  }else{
    console.log("data1..",data1);
  }
});
```

delete cognito user query.

```
const params = {
  UserPoolId : "eu-west-1_MRerlcS4o",
  Username: username
}

cognitoidentityserviceprovider.adminDeleteUser(params,(err,data) =>{
  if(err){
    console.log("err.",err);
    reject(err)
  }else{
    console.log("data..",data);
    resolve(data)
  }
});
```

delete user from dynamodb

```
var params1 = {
  TableName:"Users",
  FilterExpression : "#user_status.#Email = :email",
  ExpressionAttributeNames:{
    "#user_status": "Users",
    "#Email":"Email",
  },
  ExpressionAttributeValues:{
    ":email" : email
  }
};

const findUser = await docClient.scan(params1).promise();

if(findUser.Items !== undefined){
  findUser.Items.forEach(async el =>{
    const params2 = {
      TableName : "Users",
      Key : {
        UserID : el.UserID
      }
    }
    const deleteItem = await docClient.delete(params2).promise();
  })
}
```

<https://www.tabnine.com/code/javascript/functions/amazon-cognito-identity-js/CognitoUser/authenticateUser>

<https://medium.com/@prasadjay/amazon-cognito-user-pools-in-nodejs-as-fast-as-possible-22d586c5c8ec>