**CreateTenantReference**

'use strict';

var AWS = require('aws-sdk'),

uuid = require('uuid'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.writeTenantReference = function(event, context, callback){

var scanParams = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(scanParams, function(err, data){

if(err){

callback(err, null);

}

else{

//find max + 1

var max = 0;

for(var i = 0; i < data.Items.length; i++) {

if(data.Items[i].tenant\_reference\_id > max) {

max = data.Items[i].tenant\_reference\_id;

}

}

max += 1;

//store max in property id

var params = {

Item : {

"tenant\_reference\_id" : max,

"tenant\_id" : event.tenant\_id,

"owner\_id" : event.owner\_id,

"first\_name" : event.first\_name,

"last\_name" : event.last\_name,

"address\_line\_1" : event.address\_line\_1,

"address\_line\_2" : event.address\_line\_2,

"city" : event.city,

"state" : event.state,

"zip" : event.zip,

"county\_district" : event.county\_district,

"contact\_phone" : event.contact\_phone,

"contact\_email" : event.contact\_email,

"primary\_contact" : event.primary\_contact

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

callback(null, data.Item);

}

});

var params = {

Item : {

"tenant\_reference\_id" : event.tenant\_reference\_id,

"tenant\_id" : event.tenant\_id,

"owner\_id" : event.owner\_id,

"first\_name" : event.first\_name,

"last\_name" : event.last\_name,

"address\_line\_1" : event.address\_line\_1,

"address\_line\_2" : event.address\_line\_2,

"city" : event.city,

"state" : event.state,

"zip" : event.zip,

"county\_district" : event.county\_district,

"contact\_phone" : event.contact\_phone,

"contact\_email" : event.contact\_email,

"primary\_contact" : event.primary\_contact

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

}

**CreateTenantOccupation**

'use strict';

var AWS = require('aws-sdk'),

uuid = require('uuid'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.writeTenantOccupation = function(event, context, callback){

var scanParams = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(scanParams, function(err, data){

if(err){

callback(err, null);

}

else{

//find max + 1

var max = 0;

for(var i = 0; i < data.Items.length; i++) {

if(data.Items[i].tenant\_occupation\_id > max) {

max = data.Items[i].tenant\_occupation\_id;

}

}

max += 1;

//store max in property id

var params = {

Item : {

"tenant\_occupation\_id" : max,

"tenant\_id" : event.tenant\_id,

"owner\_id" : event.owner\_id,

"employer" : event.employer,

"title" : event.title

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

callback(null, data.Item);

}

});

var params = {

Item : {

"tenant\_occupation\_id" : event.tenant\_occupation\_id,

"tenant\_id" : event.tenant\_id,

"owner\_id" : event.owner\_id,

"employer" : event.employer,

"title" : event.title

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

}

**CreatePropertyTax**

'use strict';

var AWS = require('aws-sdk'),

uuid = require('uuid'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.writePropertyTax = function(event, context, callback){

var scanParams = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(scanParams, function(err, data){

if(err){

callback(err, null);

}

else{

//find max + 1

var max = 0;

for(var i = 0; i < data.Items.length; i++) {

if(data.Items[i].property\_tax\_id > max) {

max = data.Items[i].property\_tax\_id;

}

}

max += 1;

//store max in property id

var params = {

Item : {

"property\_tax\_id" : max,

"property\_id" : event.property\_id,

"owner\_id" : event.owner\_id,

"year" : event.year,

"year" : event.year\_paid,

"annual\_tax" : event.annual\_tax

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

callback(null, data.Item);

}

});

var params = {

Item : {

"property\_tax\_id" : event.property\_tax\_id,

"property\_id" : event.property\_id,

"owner\_id" : event.owner\_id,

"year" : event.year,

"year" : event.year\_paid,

"annual\_tax" : event.annual\_tax

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

}

**CreatePropertyMaintainanceExpense**

'use strict';

var AWS = require('aws-sdk'),

uuid = require('uuid'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.writePropertyMaintenanceExpense = function(event, context, callback){

var scanParams = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(scanParams, function(err, data){

if(err){

callback(err, null);

}

else{

//find max + 1

var max = 0;

for(var i = 0; i < data.Items.length; i++) {

if(data.Items[i].Property\_maintenance\_expense\_id > max) {

max = data.Items[i].Property\_maintenance\_expense\_id;

}

}

max += 1;

//store max in property id

var params = {

Item : {

"Property\_maintenance\_expense\_id" : max,

"expense\_amount" : event.expense\_amount,

"maintenance\_expense\_id" : event.maintenance\_expense\_id,

"owner\_id" : event.owner\_id,

"property\_id" : event.property\_id,

"receipt\_copy" : event.receipt\_copy,

"receipt\_date" : event.receipt\_date

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

callback(null, data.Item);

}

});

var params = {

Item : {

"Property\_maintenance\_expense\_id" : event.Property\_maintenance\_expense\_id,

"expense\_amount" : event.expense\_amount,

"maintenance\_expense\_id" : event.maintenance\_expense\_id,

"owner\_id" : event.owner\_id,

"property\_id" : event.property\_id,

"receipt\_copy" : event.receipt\_copy,

"receipt\_date" : event.receipt\_date

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

}

**CreateTenantBackgroundCheck**

'use strict';

var AWS = require('aws-sdk'),

uuid = require('uuid'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.writeTenantBackgroundCheck = function(event, context, callback){

var scanParams = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(scanParams, function(err, data){

if(err){

callback(err, null);

}

else{

//find max + 1

var max = 0;

for(var i = 0; i < data.Items.length; i++) {

if(data.Items[i].Tenant\_background\_check\_id > max) {

max = data.Items[i].Tenant\_background\_check\_id;

}

}

max += 1;

//store max in property id

var params = {

Item : {

"Tenant\_background\_check\_id" : max,

"tenant\_id" : event.tenant\_id,

"owner\_id" : event.owner\_id,

"criminal\_passed" : event.criminal\_passed,

"credit\_passed\_YN" : event.credit\_passed\_YN,

"eviction\_passed\_YN" : event.eviction\_passed\_YN,

"recommendation": event.recommendation

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

callback(null, data.Item);

}

});

var params = {

Item : {

"Tenant\_background\_check\_id" : event.Tenant\_background\_check\_id,

"tenant\_id" : event.tenant\_id,

"owner\_id" : event.owner\_id,

"criminal\_passed" : event.criminal\_passed,

"credit\_passed\_YN" : event.credit\_passed\_YN,

"eviction\_passed\_YN" : event.eviction\_passed\_YN,

"recommendation": event.recommendation

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

}

**CreatePropertyOwner**

'use strict';

var AWS = require('aws-sdk'),

uuid = require('uuid'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.writePropertyOwner = function(event, context, callback){

var scanParams = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(scanParams, function(err, data){

if(err){

callback(err, null);

}

else{

//find max + 1

var max = 0;

for(var i = 0; i < data.Items.length; i++) {

if(data.Items[i].owner\_id > max) {

max = data.Items[i].owner\_id;

}

}

max += 1;

//store max in property id

var params = {

Item : {

"owner\_id" : max,

"owner\_name" : event.owner\_name,

"password" : event.password,

"created\_date" : event.created\_date,

"email" : event.email,

"activated\_YN" : event.activated\_YN,

"profile\_memo": event.profile\_memo

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

callback(null, data.Item);

}

});

var params = {

Item : {

"owner\_id" : event.owner\_id,

"owner\_name" : event.owner\_name,

"password" : event.password,

"created\_date" : event.created\_date,

"email" : event.email,

"activated\_YN" : event.activated\_YN,

"profile\_memo": event.profile\_memo

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

}

**CreateRentalAgreement**

'use strict';

var AWS = require('aws-sdk'),

uuid = require('uuid'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.writeRentalAgreement = function(event, context, callback){

var scanParams = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(scanParams, function(err, data){

if(err){

callback(err, null);

}

else{

//find max + 1

var max = 0;

for(var i = 0; i < data.Items.length; i++) {

if(data.Items[i].rental\_agreement\_id > max) {

max = data.Items[i].rental\_agreement\_id;

}

}

max += 1;

//store max in property id

var params = {

Item : {

"rental\_agreement\_id" : max,

"tenant\_id" : event.tenant\_id,

"owner\_id" : event.owner\_id,

"start\_date" : event.start\_date,

"end\_date" : event.end\_date,

"monthly\_rent" : event.monthly\_rent,

"monthly\_rent\_due" : event.monthly\_rent\_due,

"advance": event.advance,

"number\_of\_occupants": event.number\_of\_occupants,

"renewal\_of\_first\_term": event.renewal\_of\_first\_term,

"rental\_agreement\_copy": event.rental\_agreement\_copy,

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

callback(null, data.Item);

}

});

var params = {

Item : {

"rental\_agreement\_id" : event.rental\_agreement\_id,

"tenant\_id" : event.tenant\_id,

"owner\_id" : event.owner\_id,

"start\_date" : event.start\_date,

"end\_date" : event.end\_date,

"monthly\_rent" : event.monthly\_rent,

"monthly\_rent\_due" : event.monthly\_rent\_due,

"advance": event.advance,

"number\_of\_occupants": event.number\_of\_occupants,

"renewal\_of\_first\_term": event.renewal\_of\_first\_term,

"rental\_agreement\_copy": event.rental\_agreement\_copy,

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

}

**CreateTenant**

'use strict';

var AWS = require('aws-sdk'),

uuid = require('uuid'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.writeTenant = function(event, context, callback){

var scanParams = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(scanParams, function(err, data){

if(err){

callback(err, null);

}

else{

//find max + 1

var max = 0;

for(var i = 0; i < data.Items.length; i++) {

if(data.Items[i].tenant\_id > max) {

max = data.Items[i].tenant\_id;

}

}

max += 1;

//store max in property id

var params = {

Item : {

"tenant\_id" : max,

"property\_id" : event.property\_id,

"owner\_id" : event.owner\_id,

"first\_name" : event.first\_name,

"middle\_name" : event.middle\_name,

"last\_name" : event.last\_name,

"age" : event.age,

"contact\_email" : event.contact\_email,

"contact\_phone" : event.contact\_phone,

"primary\_contact" : event.primary\_contact

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

callback(null, data.Item);

}

});

var params = {

Item : {

"tenant\_id" : event.tenant\_id,

"property\_id" : event.property\_id,

"owner\_id" : event.owner\_id,

"first\_name" : event.first\_name,

"middle\_name" : event.middle\_name,

"last\_name" : event.last\_name,

"age" : event.age,

"contact\_email" : event.contact\_email,

"contact\_phone" : event.contact\_phone,

"primary\_contact" : event.primary\_contact

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

}

**CreatePurchaseDetails**

'use strict';

var AWS = require('aws-sdk'),

uuid = require('uuid'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.writePurchaseDetails = function(event, context, callback){

var scanParams = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(scanParams, function(err, data){

if(err){

callback(err, null);

}

else{

//find max + 1

var max = 0;

for(var i = 0; i < data.Items.length; i++) {

if(data.Items[i].purchase\_id > max) {

max = data.Items[i].purchase\_id;

}

}

max += 1;

//store max in property id

var params = {

Item : {

"purchase\_id" : max,

"owner\_id" : event.owner\_id,

"price" : event.price,

"property\_id" : event.property\_id,

"year" : event.year

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

callback(null, data.Item);

}

});

var params = {

Item : {

"purchase\_id" : event.purchase\_id,

"owner\_id" : event.owner\_id,

"price" : event.price,

"property\_id" : event.property\_id,

"year" : event.year

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

}

**CreateProperty**

'use strict';

var AWS = require('aws-sdk'),

uuid = require('uuid'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.writeProperty = function(event, context, callback){

var scanParams = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(scanParams, function(err, data){

if(err){

callback(err, null);

}

else{

//find max + 1

var max = 0;

for(var i = 0; i < data.Items.length; i++) {

if(data.Items[i].property\_id > max) {

max = data.Items[i].property\_id;

}

}

max += 1;

//store max in property id

var params = {

Item : {

"property\_id" : max,

"address\_Line\_1" : event.address\_line\_1,

"address\_line\_2" : event.address\_line\_2,

"city" : event.city,

"state" : event.state,

"zip" : event.zip

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

callback(null, data.Item);

}

});

var params = {

Item : {

"property\_id" : event.property\_id,

"address\_Line\_1" : event.address\_line\_1,

"address\_line\_2" : event.address\_line\_2,

"city" : event.city,

"state" : event.state,

"zip" : event.zip

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

}

**CreateMaintainenceExpenseType**

'use strict';

var AWS = require('aws-sdk'),

uuid = require('uuid'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.writeExpenseType = function(event, context, callback){

var scanParams = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(scanParams, function(err, data){

if(err){

callback(err, null);

}

else{

//find max + 1

var max = 0;

for(var i = 0; i < data.Items.length; i++) {

if(data.Items[i].maintenance\_expense\_id > max) {

max = data.Items[i].maintenance\_expense\_id;

}

}

max += 1;

//store max in property id

var params = {

Item : {

"maintenance\_expense\_id" : max,

"description" : event.description,

"owner\_id" : event.owner\_id,

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

callback(null, data.Item);

}

});

var params = {

Item : {

"maintenance\_expense\_id" : event.maintenance\_expense\_id,

"description" : event.description,

"owner\_id" : event.owner\_id,

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

}

**CreatePropertyMortgageLoan**

'use strict';

var AWS = require('aws-sdk'),

uuid = require('uuid'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.writeMortgageLoan = function(event, context, callback){

var scanParams = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(scanParams, function(err, data){

if(err){

callback(err, null);

}

else{

//find max + 1

var max = 0;

for(var i = 0; i < data.Items.length; i++) {

if(data.Items[i].Property\_mortgage\_loan\_id > max) {

max = data.Items[i].Property\_mortgage\_loan\_id;

}

}

max += 1;

//store max in property id

var params = {

Item : {

"Property\_mortgage\_loan\_id" : max,

"monthly\_mortgage\_amount" : event.monthly\_mortgage\_amount,

"mortgage\_lender" : event.mortgage\_lender,

"outstanding\_balance" : event.outstanding\_balance,

"owner\_id" : event.owner\_id,

"property\_id" : event.property\_id

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

callback(null, data.Item);

}

});

var params = {

Item : {

"Property\_mortgage\_loan\_id" : event.Property\_mortgage\_loan\_id,

"monthly\_mortgage\_amount" : event.monthly\_mortgage\_amount,

"mortgage\_lender" : event.mortgage\_lender,

"outstanding\_balance" : event.outstanding\_balance,

"owner\_id" : event.owner\_id,

"property\_id" : event.property\_id

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

}

**CreateGeneralExpenses**

'use strict';

var AWS = require('aws-sdk'),

uuid = require('uuid'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.writeGeneralExpenses = function(event, context, callback){

var scanParams = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(scanParams, function(err, data){

if(err){

callback(err, null);

}

else{

//find max + 1

var max = 0;

for(var i = 0; i < data.Items.length; i++) {

if(data.Items[i].property\_general\_expense\_id > max) {

max = data.Items[i].property\_general\_expense\_id;

}

}

max += 1;

//store max in property id

var params = {

Item : {

"property\_general\_expense\_id" : max,

"monthly\_condo\_fee" : event.monthly\_condo\_fee,

"monthly\_home\_fee" : event.monthly\_home\_fee,

"owner\_id" : event.owner\_id,

"property\_id" : event.property\_id,

"receipt\_copy" : event.receipt\_copy,

"yearly\_property\_insurance" : event.yearly\_property\_insurance

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

callback(null, data.Item);

}

});

var params = {

Item : {

"property\_general\_expense\_id" : event.property\_general\_expense\_id,

"monthly\_condo\_fee" : event.monthly\_condo\_fee,

"monthly\_home\_fee" : event.monthly\_home\_fee,

"owner\_id" : event.owner\_id,

"property\_id" : event.property\_id,

"receipt\_copy" : event.receipt\_copy,

"yearly\_property\_insurance" : event.yearly\_property\_insurance

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

}

**CreateTenantIncomeExpense**

'use strict';

var AWS = require('aws-sdk'),

uuid = require('uuid'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.writeTenantIncomeExpense = function(event, context, callback){

var scanParams = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(scanParams, function(err, data){

if(err){

callback(err, null);

}

else{

//find max + 1

var max = 0;

for(var i = 0; i < data.Items.length; i++) {

if(data.Items[i].tenant\_income\_expense\_id > max) {

max = data.Items[i].tenant\_income\_expense\_id;

}

}

max += 1;

//store max in property id

var params = {

Item : {

"tenant\_income\_expense\_id" : max,

"tenant\_id" : event.tenant\_id,

"owner\_id" : event.owner\_id,

"monthly\_income" : event.monthly\_income,

"monthly\_expenses" : event.monthly\_expenses,

"peoperty\_owner\_owner\_id" : event.peoperty\_owner\_owner\_id

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

callback(null, data.Item);

}

});

var params = {

Item : {

"tenant\_income\_expense\_id" : event.tenant\_income\_expense\_id,

"tenant\_id" : event.tenant\_id,

"owner\_id" : event.owner\_id,

"monthly\_income" : event.monthly\_income,

"monthly\_expenses" : event.monthly\_expenses,

"peoperty\_owner\_owner\_id" : event.peoperty\_owner\_owner\_id

},

TableName : process.env.TABLE\_NAME

};

documentClient.put(params, function(err, data){

callback(err, data);

});

}

**ListBackgroundCheck**

'use strict';

var AWS = require('aws-sdk'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.listBackgroundCheck = function(event, context, callback){

var params = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(params, function(err, data){

if(err){

callback(err, null);

}else{

callback(null, data.Items);

}

});

}

**ListPurchaseDetails**

'use strict';

var AWS = require('aws-sdk'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.listPurchaseDetails = function(event, context, callback){

var params = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(params, function(err, data){

if(err){

callback(err, null);

}else{

callback(null, data.Items);

}

});

}

**ListTenant**

'use strict';

var AWS = require('aws-sdk'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.listTenant = function(event, context, callback){

var params = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(params, function(err, data){

if(err){

callback(err, null);

}else{

callback(null, data.Items);

}

});

}

**ListRentalAgreement**

'use strict';

var AWS = require('aws-sdk'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.listRentalAgreement = function(event, context, callback){

var params = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(params, function(err, data){

if(err){

callback(err, null);

}else{

callback(null, data.Items);

}

});

}

**ListPropertyGeneralExpenses**

'use strict';

var AWS = require('aws-sdk'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.listGeneralExpenses = function(event, context, callback){

var params = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(params, function(err, data){

if(err){

callback(err, null);

}else{

callback(null, data.Items);

}

});

}

**ListPropertyMaintainenceExpense**

'use strict';

var AWS = require('aws-sdk'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.listPropertyMaintenanceExpense = function(event, context, callback){

var params = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(params, function(err, data){

if(err){

callback(err, null);

}else{

callback(null, data.Items[0].Property\_maintenance\_expense\_id);

}

});

}

**ListTenantReference**

'use strict';

var AWS = require('aws-sdk'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.listTenantReference = function(event, context, callback){

var params = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(params, function(err, data){

if(err){

callback(err, null);

}else{

callback(null, data.Items);

}

});

}

**ListTenantOccupation**

'use strict';

var AWS = require('aws-sdk'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.listTenantOccupation = function(event, context, callback){

var params = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(params, function(err, data){

if(err){

callback(err, null);

}else{

callback(null, data.Items);

}

});

}

**ListProperties**

'use strict';

var AWS = require('aws-sdk'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.listProperties = function(event, context, callback){

var params = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(params, function(err, data){

if(err){

callback(err, null);

}else{

callback(null, data.Items);

}

});

}

**ListTenantIncomeExpense**

'use strict';

var AWS = require('aws-sdk'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.listTenantIncomeExpenses = function(event, context, callback){

var params = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(params, function(err, data){

if(err){

callback(err, null);

}else{

callback(null, data.Items);

}

});

}

**ListPropertyMortagageLoan**

'use strict';

var AWS = require('aws-sdk'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.listPropertyMortgageLoan = function(event, context, callback){

var params = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(params, function(err, data){

if(err){

callback(err, null);

}else{

callback(null, data.Items);

}

});

}

**ListMaintanenceExpenseType**

'use strict';

var AWS = require('aws-sdk'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.listMaintenanceExpenseTypes = function(event, context, callback){

var params = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(params, function(err, data){

if(err){

callback(err, null);

}else{

callback(null, data.Items);

}

});

}

**ListPropertyTax**

'use strict';

var AWS = require('aws-sdk'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.listPropertyTax = function(event, context, callback){

var params = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(params, function(err, data){

if(err){

callback(err, null);

}else{

callback(null, data.Items);

}

});

}

**ListPropertyOwner**

'use strict';

var AWS = require('aws-sdk'),

documentClient = new AWS.DynamoDB.DocumentClient();

exports.listPropertyOwner = function(event, context, callback){

var params = {

TableName : process.env.TABLE\_NAME

};

documentClient.scan(params, function(err, data){

if(err){

callback(err, null);

}else{

callback(null, data.Items);

}

});

}

**UpdatePurchaseDetails**

'use strict';

var AWS = require("aws-sdk");

var docClient = new AWS.DynamoDB.DocumentClient()

exports.updatePurchaseDetails = function(data, context, callback){

var params = {

TableName:process.env.TABLE\_NAME,

Key:{

"purchase\_id":data.purchase\_id

},

UpdateExpression: "set owner\_id = :oid, price = :price, property\_id= :pid, year = :yr",

ConditionExpression: "property\_id = :property\_id",

ExpressionAttributeValues:{

":property\_id":data.purchase\_id,

":oid": data.oid,

":price": data.price,

":pid": data.pid,

":yr": data.yr

},

ReturnValues: "UPDATED\_NEW"

};

console.log("Attempting a conditional update for property...");

docClient.update(params,function(err,data){

if(err){

console.error("Unable to update property record. Error JSON:", JSON.stringify(err,null,2));

}

else{

console.log("Successful Update!");

}

});

}

**UpdateTenantIncomeExpense**

'use strict';

var AWS = require("aws-sdk");

var docClient = new AWS.DynamoDB.DocumentClient()

exports.updateTenantIncomeExpense = function(data, context, callback){

var params = {

TableName:process.env.TABLE\_NAME,

Key:{

"tenant\_income\_expense\_id":data.tenant\_income\_expense\_id

},

UpdateExpression: "set tenant\_id = :tid, owner\_id = :oid, monthly\_income = :mi, monthly\_expenses = :me, property\_owner\_owner\_id = :pooid",

ConditionExpression: "tenant\_income\_expense\_id = :tenant\_income\_expense\_id",

ExpressionAttributeValues:{

":tenant\_income\_expense\_id":data.tenant\_income\_expense\_id,

":tid":data.tid,

":oid": data.oid,

":mi":data.mi,

":me":data.me,

":pooid":data.pooid

},

ReturnValues: "UPDATED\_NEW"

};

console.log("Attempting a conditional update for tenant income exprese...");

docClient.update(params,function(err,data){

if(err){

console.error("Unable to update tenant income expense record. Error JSON:", JSON.stringify(err,null,2));

}

else{

console.log("Successful Update!!!!!!!!!!");

}

});

}

**UpdateMaintanenceExpenses**

'use strict';

var AWS = require("aws-sdk");

var docClient = new AWS.DynamoDB.DocumentClient()

exports.updatePropertyMaintenanceExpenses = function(data, context, callback){

var params = {

TableName:process.env.TABLE\_NAME,

Key:{

"Property\_maintenance\_expense\_id":data.Property\_maintenance\_expense\_id

},

UpdateExpression: "set expense\_amount = :exp, owner\_id = :ownid, property\_id = :propid, receipt\_copy = :rct, receipt\_date = :rct\_date",

ConditionExpression: "Property\_maintenance\_expense\_id = :Property\_maintenance\_expense\_id",

ExpressionAttributeValues:{

":Property\_maintenance\_expense\_id":data.Property\_maintenance\_expense\_id ,

":exp": data.expense\_amount,

":ownid": data.owner\_id,

":propid": data.property\_id,

":rct": data.receipt\_copy,

":rct\_date": data.receipt\_date

},

ReturnValues: "UPDATED\_NEW"

};

console.log("Attempting a conditional update for property...");

docClient.update(params,function(err,data){

if(err){

console.error("Unable to update property record. Error JSON:", JSON.stringify(err,null,2));

}

else{

console.log("Successful Update!");

}

});

}

**UpdateTenantBackgroundCheck**

'use strict';

var AWS = require("aws-sdk");

var docClient = new AWS.DynamoDB.DocumentClient()

exports.updateBackgroundCheck = function(data, context, callback){

var params = {

TableName:process.env.TABLE\_NAME,

Key:{

"Tenant\_background\_check\_id":data.Tenant\_background\_check\_id

},

UpdateExpression: "set tenant\_id = :tid, owner\_id = :oid, criminal\_passed = :cp, credit\_passed\_YN = :creditp, eviction\_passed\_YN = :ep, recommendation = :rec",

ConditionExpression: "property\_id = :property\_id",

ExpressionAttributeValues:{

":Tenant\_background\_check\_id":data.Tenant\_background\_check\_id,

":tid":data.tid,

":oid": data.oid,

":cp": data.cp,

":creditp": data.credit,

":ep": data.ep,

":rec": data.rec

},

ReturnValues: "UPDATED\_NEW"

};

console.log("Attempting a conditional update for property...");

docClient.update(params,function(err,data){

if(err){

console.error("Unable to update property record. Error JSON:", JSON.stringify(err,null,2));

}

else{

console.log("Successful Update!");

}

});

}

**UpdateTenantOccupation**

'use strict';

var AWS = require("aws-sdk");

var docClient = new AWS.DynamoDB.DocumentClient()

exports.updateTenantOccupation = function(data, context, callback){

var params = {

TableName:process.env.TABLE\_NAME,

Key:{

"tenant\_occupation\_id":data.tenant\_occupation\_id

},

UpdateExpression: "set tenant\_id = :tid, owner\_id = :oid, employer = :epy, title = :tle",

ConditionExpression: "tenant\_occupation\_id = :tenant\_occupation\_id",

ExpressionAttributeValues:{

":tenant\_occupation\_id":data.tenant\_occupation\_id,

":tid":data.tid,

":oid": data.oid,

":epy":data.epy,

":tle":data.tle

},

ReturnValues: "UPDATED\_NEW"

};

console.log("Attempting a conditional update for tenant...");

docClient.update(params,function(err,data){

if(err){

console.error("Unable to update tenant record. Error JSON:", JSON.stringify(err,null,2));

}

else{

console.log("Successful Update!!!!!!!!!!");

}

});

}

**UpdatePropertyMortagageLoab**

'use strict';

var AWS = require("aws-sdk");

var docClient = new AWS.DynamoDB.DocumentClient()

exports.updatePropertyMortgageLoan = function(data, context, callback){

var params = {

TableName:process.env.TABLE\_NAME,

Key:{

"Property\_mortgage\_loan\_id":data.Property\_mortgage\_loan\_id

},

UpdateExpression: "set monthly\_mortgage\_amount = :mma, owner\_id = :ownid, property\_id = :propid, mortgage\_lender = :ml, outstanding\_balance = :outb",

ConditionExpression: "Property\_mortgage\_loan\_id = :Property\_mortgage\_loan\_id",

ExpressionAttributeValues:{

":Property\_mortgage\_loan\_id":data.Property\_mortgage\_loan\_id ,

":mma": data.monthly\_mortgage\_amount,

":ownid": data.owner\_id,

":propid": data.property\_id,

":ml": data.mortgage\_lender,

":outb": data.outstanding\_balance

},

ReturnValues: "UPDATED\_NEW"

};

console.log("Attempting a conditional update for property...");

docClient.update(params,function(err,data){

if(err){

console.error("Unable to update property record. Error JSON:", JSON.stringify(err,null,2));

}

else{

console.log("Successful Update!");

}

});

}

**UpdateTenant**

'use strict';

var AWS = require("aws-sdk");

var docClient = new AWS.DynamoDB.DocumentClient()

exports.updateTenant = function(data, context, callback){

var params = {

TableName:process.env.TABLE\_NAME,

Key:{

"tenant\_id":data.tenant\_id

},

UpdateExpression: "set property\_id = :pid, owner\_id = :oid, first\_name = :fn, last\_name = :ln, age = :age, contact\_email = :ce, contact\_phone = :cp",

ConditionExpression: "tenant\_id = :tenant\_id",

ExpressionAttributeValues:{

":tenant\_id":data.tenant\_id,

":pid":data.pid,

":oid": data.oid,

":fn": data.fn,

":ln": data.ln,

":age": data.age,

":ce": data.ce,

":cp": data.cp

},

ReturnValues: "UPDATED\_NEW"

};

console.log("Attempting a conditional update for tenant...");

docClient.update(params,function(err,data){

if(err){

console.error("Unable to update tenant record. Error JSON:", JSON.stringify(err,null,2));

}

else{

console.log("Successful Update!!!!!!!!!!");

}

});

}

**UpdatePropertyTax**

'use strict';

var AWS = require("aws-sdk");

var docClient = new AWS.DynamoDB.DocumentClient()

exports.updatePropertyTax = function(data, context, callback){

var params = {

TableName:process.env.TABLE\_NAME,

Key:{

"property\_tax\_id":data.property\_tax\_id

},

UpdateExpression: "set property\_id = :pd, owner\_id = :oid, year = :yr, year\_paid = :yrp, annual\_tax = :at",

ConditionExpression: "rental\_agreement\_id = :rental\_agreement\_id",

ExpressionAttributeValues:{

":property\_tax\_id":data.property\_tax\_id,

":tid":data.pd,

":oid": data.oid,

":year": data.yr,

":year\_paid": data.yrp,

":annual\_tax": data.at

},

ReturnValues: "UPDATED\_NEW"

};

console.log("Attempting a conditional update for property...");

docClient.update(params,function(err,data){

if(err){

console.error("Unable to update property record. Error JSON:", JSON.stringify(err,null,2));

}

else{

console.log("Successful Update!");

}

});

}

**UpdatePropertyGeneralExpense**

'use strict';

var AWS = require("aws-sdk");

var docClient = new AWS.DynamoDB.DocumentClient()

exports.updatePropertyGeneralExpenses = function(data, context, callback){

var params = {

TableName:process.env.TABLE\_NAME,

Key:{

"property\_general\_expense\_id":data.property\_general\_expense\_id

},

UpdateExpression: "set monthly\_condo\_fee = :mcf, owner\_id = :ownid, monthly\_home\_fee = :mhf, property\_id = :propid, receipt\_copy = :rct, yearly\_property\_insurance = :ypi",

ConditionExpression: "property\_general\_expense\_id = :property\_general\_expense\_id",

ExpressionAttributeValues:{

":property\_general\_expense\_id":data.property\_general\_expense\_id ,

":mcf": data.monthly\_condo\_fee,

":ownid": data.ownid,

":mhf": data.monthly\_home\_fee,

":propid": data.property\_id,

":rct": data.receipt\_copy,

":ypi": data.yearly\_property\_insurance

},

ReturnValues: "UPDATED\_NEW"

};

console.log("Attempting a conditional update for property...");

docClient.update(params,function(err,data){

if(err){

console.error("Unable to update property record. Error JSON:", JSON.stringify(err,null,2));

}

else{

console.log("Successful Update!!!!!!!!!!!!!!!!!!!!");

}

});

}

**UpdateMaintanenceExpenseType**

'use strict';

var AWS = require("aws-sdk");

var docClient = new AWS.DynamoDB.DocumentClient()

exports.updateMaintenanceExpenseType = function(data, context, callback){

var params = {

TableName:process.env.TABLE\_NAME,

Key:{

"maintenance\_expense\_id":data.maintenance\_expense\_id

},

UpdateExpression: "set description = :desc, owner\_id = :ownid",

ConditionExpression: "maintenance\_expense\_id = :maintenance\_expense\_id",

ExpressionAttributeValues:{

":maintenance\_expense\_id":data.maintenance\_expense\_id ,

":desc": data.desc,

":ownid": data.ownid,

},

ReturnValues: "UPDATED\_NEW"

};

console.log("Attempting a conditional update for property...");

docClient.update(params,function(err,data){

if(err){

console.error("Unable to update property record. Error JSON:", JSON.stringify(err,null,2));

}

else{

console.log("Successful Update!");

}

});

}

**UpdateTenantReference**

'use strict';

var AWS = require("aws-sdk");

var docClient = new AWS.DynamoDB.DocumentClient()

exports.updateTenantReference = function(data, context, callback){

var params = {

TableName:process.env.TABLE\_NAME,

Key:{

"tenant\_reference\_id":data.tenant\_reference\_id

},

UpdateExpression: "set tenant\_id = :tid, owner\_id = :oid, first\_name = :fn, last\_name = :ln, address\_line\_1 = :al1, address\_line\_2 = :al2, city = :ct, st = :st, zip = :zp, county\_or\_district = :cd",

ConditionExpression: "tenant\_reference\_id = :tenant\_reference\_id",

ExpressionAttributeValues:{

":tenant\_reference\_id":data.tenant\_reference\_id,

":tid":data.tid,

":oid": data.oid,

":fn": data.fn,

":ln": data.ln,

":al1":data.al1,

":al2":data.al2,

":ct":data.ct,

":st":data.st,

":zp":data.zp,

":cd":data.cd

},

ReturnValues: "UPDATED\_NEW"

};

console.log("Attempting a conditional update for tenant reference...");

docClient.update(params,function(err,data){

if(err){

console.error("Unable to update tenant reference record. Error JSON:", JSON.stringify(err,null,2));

}

else{

console.log("Successful Update!!!!!!!!!!");

}

});

}

**UpdatePropertyOwner**

'use strict';

var AWS = require("aws-sdk");

var docClient = new AWS.DynamoDB.DocumentClient()

exports.updatePropOwner = function(data, context, callback){

var params = {

TableName:process.env.TABLE\_NAME,

Key:{

"owner\_id":data.owner\_id

},

UpdateExpression: "set owner\_name = :on, password = :pw, created\_date = :cd, email = :em, activated\_YN = :ay, proflie\_memo = :pm",

ConditionExpression: "owner\_id = :owner\_id",

ExpressionAttributeValues:{

":owner\_id":data.owner\_id,

":on":data.on,

":pw":data.pw,

":cd":data.cd,

":em":data.em,

":ay":data.ay,

":pm":data.pm

},

ReturnValues: "UPDATED\_NEW"

};

console.log("Attempting a conditional update for property owner...");

docClient.update(params,function(err,data){

if(err){

console.error("Unable to update property owner record. Error JSON:", JSON.stringify(err,null,2));

}

else{

console.log("Successful Update!!!!!!!!!!");

}

});

}

**UpdateRentalAgreement**

'use strict';

var AWS = require("aws-sdk");

var docClient = new AWS.DynamoDB.DocumentClient()

exports.updateRentalAgreement = function(data, context, callback){

var params = {

TableName:process.env.TABLE\_NAME,

Key:{

"rental\_agreement\_id":data.rental\_agreement\_id

},

UpdateExpression: "set tenant\_id = :tid, owner\_id = :oid, start\_date = :sd, end\_date = :ed, monthly\_rent = :mr, monthly\_rent\_due = :mrd, advance =: adv, number\_of\_occupants = :nof, renewal\_of\_first\_term = :roft, rental\_agreement\_copy = :rac",

ConditionExpression: "rental\_agreement\_id = :rental\_agreement\_id",

ExpressionAttributeValues:{

":Tenant\_background\_check\_id":data.rental\_agreement\_id,

":tid":data.tid,

":oid": data.oid,

":sd": data.sd,

":ed": data.ed,

":mr": data.mr,

":mrd": data.mrd,

":adv": data.adv,

":nof": data.nof,

":roft": data.roft,

":rac": data.rac

},

ReturnValues: "UPDATED\_NEW"

};

console.log("Attempting a conditional update for property...");

docClient.update(params,function(err,data){

if(err){

console.error("Unable to update property record. Error JSON:", JSON.stringify(err,null,2));

}

else{

console.log("Successful Update!");

}

});

}

**UpdateProperty**

'use strict';

var AWS = require("aws-sdk");

var docClient = new AWS.DynamoDB.DocumentClient()

exports.updateProperty = function(data, context, callback){

var params = {

TableName:process.env.TABLE\_NAME,

Key:{

"property\_id":data.property\_id

},

UpdateExpression: "set address\_Line\_1 = :al1, address\_line\_2 = :al2, city = :city, st = :st, zip = :zp",

ConditionExpression: "property\_id = :property\_id",

ExpressionAttributeValues:{

":property\_id":data.property\_id ,

":al1": data.al1,

":al2": data.al2,

":city": data.city,

":st": data.st,

":zp": data.zp

},

ReturnValues: "UPDATED\_NEW"

};

console.log("Attempting a conditional update for property...");

docClient.update(params,function(err,data){

if(err){

console.error("Unable to update property record. Error JSON:", JSON.stringify(err,null,2));

}

else{

console.log("Successful Update!!!!!!!!!!!!!!!!!!!!");

}

});

}