1. Create Table

var params = {

TableName : "Movie",

KeySchema: [

{ AttributeName: "Year", KeyType: "HASH" }, //Partition key

{ AttributeName: "Title", KeyType: "RANGE" } //Sort key

],

AttributeDefinitions: [

{ AttributeName: "Year", AttributeType: "N" },

{ AttributeName: "Title", AttributeType: "S" }

],

ProvisionedThroughput: {

ReadCapacityUnits: 1,

WriteCapacityUnits: 1

}

};

dynamodb.createTable(params, function(err, data) {

if (err)

console.log(JSON.stringify(err, null, 2));

else

console.log(JSON.stringify(data, null, 2));

});

Result:

{

"TableDescription": {

"AttributeDefinitions": [

{

"AttributeName": "Year",

"AttributeType": "N"

},

{

"AttributeName": "Title",

"AttributeType": "S"

}

],

"TableName": "Movie",

"KeySchema": [

{

"AttributeName": "Year",

"KeyType": "HASH"

},

{

"AttributeName": "Title",

"KeyType": "RANGE"

}

],

"TableStatus": "ACTIVE",

"CreationDateTime": "2016-03-26T01:28:40.790Z",

"ProvisionedThroughput": {

"LastIncreaseDateTime": "1970-01-01T00:00:00.000Z",

"LastDecreaseDateTime": "1970-01-01T00:00:00.000Z",

"NumberOfDecreasesToday": 0,

"ReadCapacityUnits": 1,

"WriteCapacityUnits": 1

},

"TableSizeBytes": 0,

"ItemCount": 0,

"TableArn": "arn:aws:dynamodb:ddblocal:000000000000:table/Movie"

}

}

1. Table Description

var params = {

TableName: "Movie"

};

dynamodb.describeTable(params, function(err, data) {

if (err)

console.log(JSON.stringify(err, null, 2));

else

console.log(JSON.stringify(data, null, 2));

});

1. List Tables

var params = {};

dynamodb.listTables(params, function(err, data) {

if (err)

console.log(JSON.stringify(err, null, 2));

else

console.log(JSON.stringify(data, null, 2));

});

Result:

{

"TableNames": [

"Movie",

"Music"

]

}

1. Write Items

var params = {

TableName: "Movie",

Item: {

"Year":2013,

"Title":"Turn It Down, Or Else!",

"directors" : [

"Alice Smith",

"Bob Jones"

],

"release\_date" : "2013-01-18T00:00:00Z",

"rating" : 6.2,

"genres" : [

"Comedy",

"Drama"

],

"image\_url" : "http://ia.media-imdb.com/images/N/O9ERWAU7FS797AJ7LU8HN09AMUP908RLlo5JF90EWR7LJKQ7@@.\_V1\_SX400\_.jpg",

"plot" : "A rock band plays their music at high volumes, annoying the neighbors.",

"rank" : 11,

"running\_time\_secs" : 5215,

"actors" : [

"David Matthewman",

"Ann Thomas",

"Jonathan G. Neff"

]

}

};

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

if (err)

console.log(JSON.stringify(err, null, 2));

else

console.log(JSON.stringify(data, null, 2));

});

Result:

{}

1. GetItem

{

"Item": {

"actors": [

"Sonam Kapoor",

"Shabana Azmi",

"Shekhar Ravjiani"

],

"Year": 2016,

"release\_date": "2016-02-19T00:00:00Z",

"plot": "Featuring a girl who saved lives during plane hijack",

"genres": [

"Drama",

"Biopic"

],

"image\_url": "http://ia.media-imdb.com/images/N/O9ERWAU7FS797AJ7LU8HN09AMUP908RLlo5JF90EWR7LJKQ7@@.\_V1\_SX400\_.jpg",

"directors": [

"Ram Madhvani"

],

"rating": 8.2,

"Title": "Neerja",

"rank": 20,

"running\_time\_secs": 5215

}

}

1. GetItem subset retrival

Retrieving only actors of the film

var params = {

TableName: "Movie",

Key: {

"Year": 2015,

"Title": "Bajirao Mastani"

},

ProjectionExpression: "actors"

};

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

if (err)

console.log(JSON.stringify(err, null, 2));

else

console.log(JSON.stringify(data, null, 2));

});

{

"Item": {

"actors": [

"Ranveer Singh",

"Deepika Padukon",

"Priyanka Chopra"

]

}

}