An initiative by women++

Hack'n'Lead

Leverage the Cloud to Deploy and Run your Backend App

Tuesday, July 4
18:00 - 19:30

Onsite workshop
Heinrichstrasse 200, 8005 Zürich

Iris Hunkeler
Hack'n'Lead Ambassador

women++ is a Swiss non-profit organization dedicated to increasing diversity in the tech industry.

+1K

Participants joining our initiatives

>80%

Female participation ratio

+50

women transitioned to tech

Our core values of leadership, education, inclusion and collaboration.

women++

Hack'n'Lead

Switzerland's first women-friendly hackathon

Studies show that in particular, women want:

safe spaces

Women actively search for women's names on the speaker panels and planning committees

to contribute

Women participate only if they feel like they have the right skillset to contribute, unlike men who participate out of interest for a given project.

to prepare

Women tend to prefer to form a team and scout the physical space in advance.

childcare

In EU-27 countries, 85% of parents reducing their working hours in favor of childcare are women.

220+
Applicants

80/20
women/men collaborating hand in hand

70%
Attended a hackathon for the first time

30
Different nationalities

44
Mentors from 18 companies

22
innovative projects built in 2 days

An initiative by women++

Hack'n'Lead

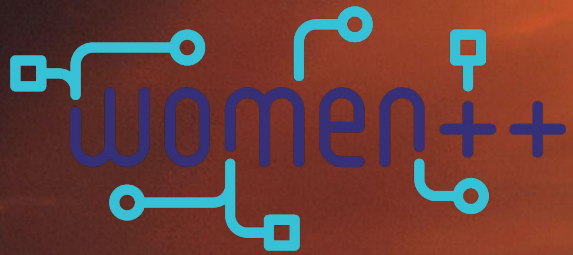
Leverage the Cloud to Deploy and Run your Backend App

Tuesday, July 4
18:00 - 19:30

Onsite workshop
Heinrichstrasse 200, 8005 Zürich

Iris Hunkeler
Hack'n'Lead Ambassador

Leverage the Cloud to Deploy and Run your Backend App



04.07.2023

Iris Hunkeler

Goals

- 🎯 Understand what the cloud is and its benefits
- 🎯 Understand the difference between serverless and provisioned resources
- 🎯 Identify use-cases for a serverless setup in the cloud
- ✍🏽 Deploy your first serverless application

Iris Hunkeler

Software Engineer / Senior Consultant
at Netlight, Zurich (Switzerland)

https://www.linkedin.com/in/iris-hunkeler

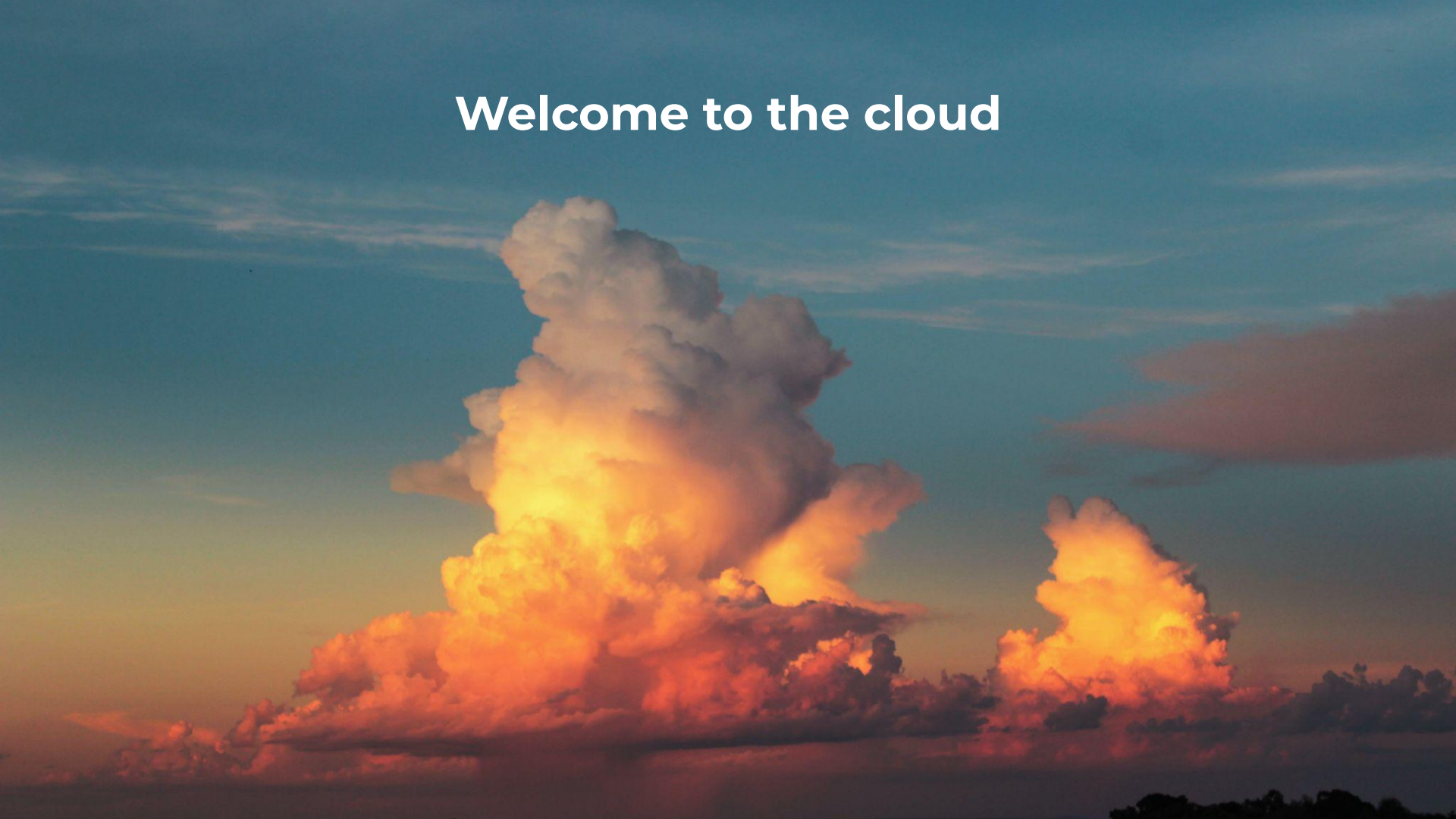
https://medium.com/@iris.hunkeler

https://github.com/iris-hunkeler

Agenda

- Cloud introduction
- Account creation and budgeting on AWS
- ✍🏾 A first, simple Lambda function
- Other helpful services on AWS:

  EventBridge, API Gateway, DynamoDB, Simple Notification Service
- ✍🏻 Build a “To Do List” app on AWS
- Review & Closing


Welcome to the cloud

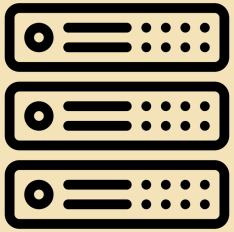
What is the cloud, actually?

Software as a Service (SaaS)

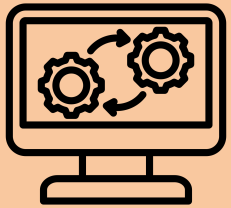
Platform as a Service (PaaS)

Infrastructure as a Service (IaaS)

Physical data center



Servers, networking, storage



Operating systems

Database management

Applications

Frontend vs. Backend

Frontend

- Runs on the device of an end-user (phone, laptop)
- Cannot share information between different devices
- User Interface

Backend

- Runs on a server, e.g. in the cloud
- Can store data and share with different users
- Data storage + Computations

The three big cloud providers

Amazon Web Services

32% market share

Microsoft Azure

23% market share

Google Cloud

10% market share

Market share information: https://www.statista.com/statistics/967365/worldwide-cloud-infrastructure-services-market-share-vendor/

Provisioned vs. Serverless

Provisioned Resources

- User defines what they need at what time
- Fixed cost
- “Pay-what-you-ordered”

→ for predictable, fairly constant usage

Serverless Resources

- Service is always available
- Variable cost
- “Pay-what-you-use” model

→ for unpredictable, irregular usage

→ good choice for prototyping!

THERE IS NO CLOUD!

It’s just someone else’s computer

Welcome to AWS

- Started in 2006
- Global set of cloud based products:

  Compute, Storage, Networking, Analytics, Security and many more.
- over 200 services
- organized into over

  20 regions

AWS Free Tier

AWS offers a generous **Free Tier** to start experimenting and exploring

Three types: Free trials, 12 month free, Always free

Examples

- **AWS Lambda**: 1 Million requests per month (always free)
- **AWS DynamoDB**: 25 GB storage (always free)
- **API Gateway**: 1 Million calls per months (12 months free)
- **Amazon SNS**: 1 Million publishes (always free)

✍🏾 Creating your AWS account

➡ https://portal.aws.amazon.com/billing/signup

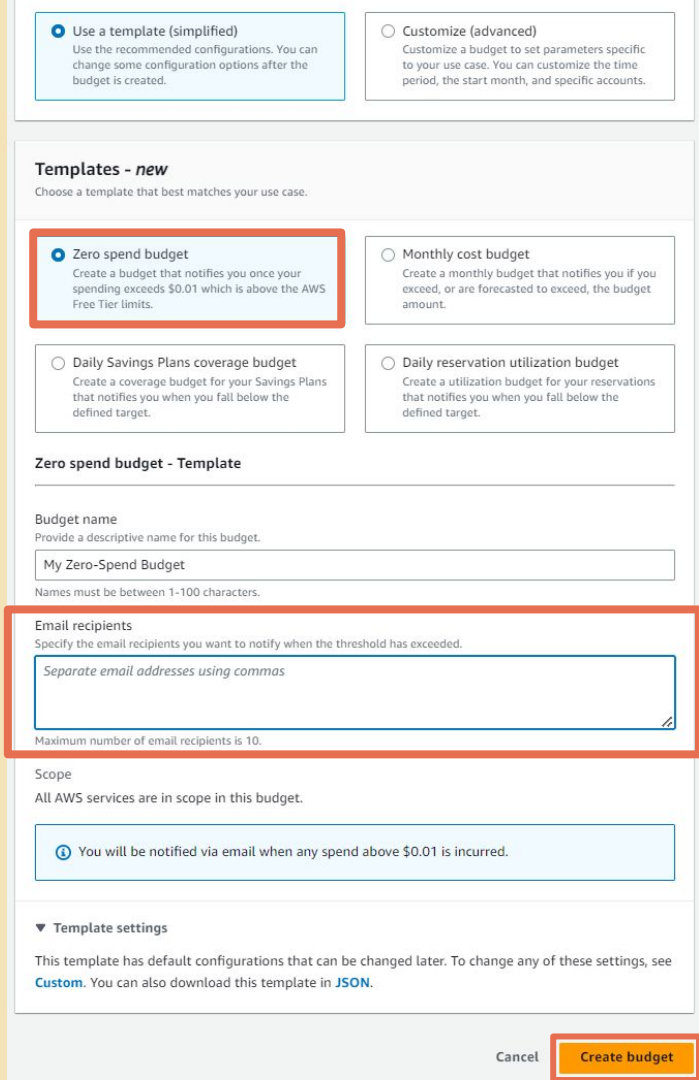
1. Email + Root Account Name (e.g. your name)
    a. Confirm Email
    b. Define Root Password
2. Enter Contact Information
3. Enter Credit Card Information
4. Confirm Identity using SMS
5. Select Free Support Plan

🎉 You can now login to AWS with your root account!

✍️ Budgeting

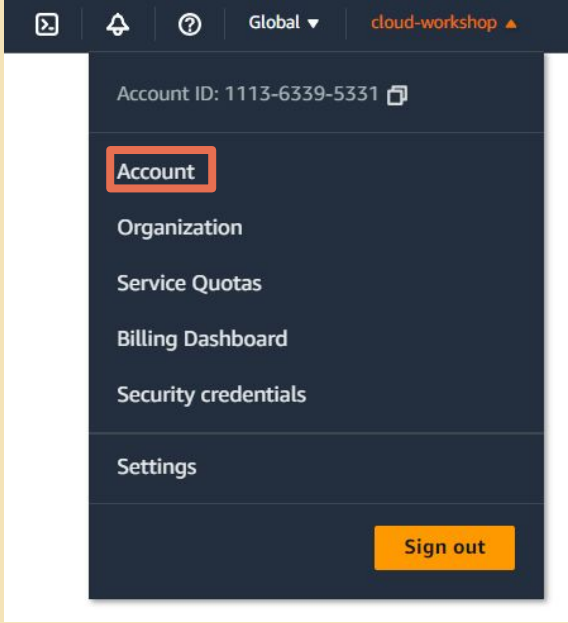
1. Login to Management Console
2. Search for “AWS Budgets”
3. Click “Create a Budget”
4. Use Template “Zero spend budget”
    - Enter Email address
5. Click “Create Budget”

💰 You will now be notified, should *any* costs occur in your AWS account



What if I want to close my account?

1. Login to the management console
2. Click on your account name on the top right → Select “Account”
3. Scroll to the very bottom and click “Close Account”



Close Account

Please review this important account closure guidance. Specifically:

- **Agreement termination**: Closing your account will serve as your notice of termination of the AWS Customer Agreement (or any other AWS agreement governing this account) for this account.
- **Billing**: You remain responsible for all outstanding fees and charges, incuding this month's usage and active subscriptions (such as Reserved Instances).
- **Reactivation**: You may reopen your AWS account for 90 days after closure. If you reopen your account, you may be charged for any active resources. After 90 days your account will be permanently closed, any remaining content will be deleted, and unused credits will be lost.
- **GovCloud**: Closing this account will also close any linked GovCloud accounts.

If you are experiencing unexpected changes, review how to troubleshoot common changes, including Free Tier changes, unwanted resources and unauthorized activity, without closing your account.

Close Account

Example - Let’s build our own To Do list

Planned functionality

- View all items
  - an item has a description and a due date
- Create a new item
- Delete an item
- Send reminder email about items due today

→ runs in the cloud, serverless setup

AWS Lambda

- **Serverless**, event-driven **compute** service
- natively supports **multiple languages**: Java, Go, PowerShell, Node.js, C#, Python, and Ruby code
- part of the **AWS Free Tier**: 1 Million requests per month (always free)

✍🏻 Practice - A first simple Lambda function

🎯 Build a Lambda that returns the “To Do item” you pass to it
(stateless = no data storage, no data base)

Resources for exercise

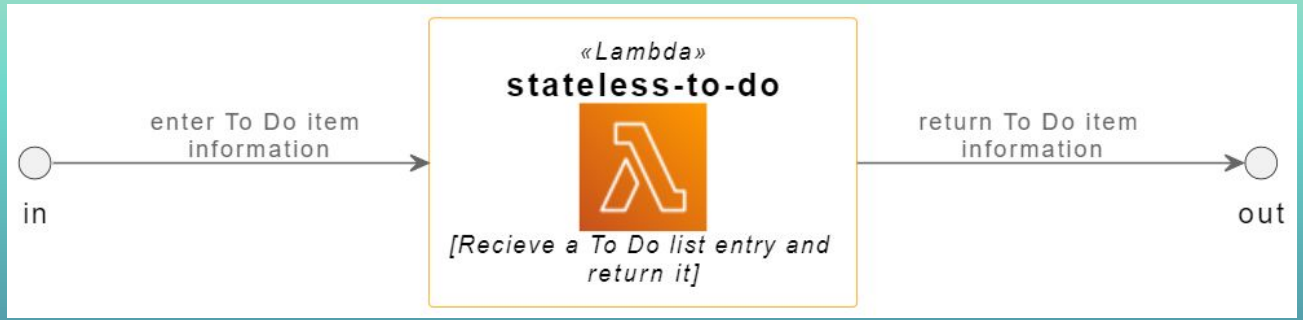
“How To on AWS” including links to source code:

https://bit.ly/howtoaws

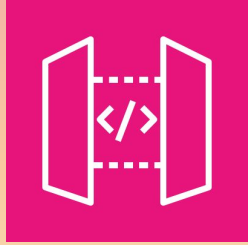
Break

Review: What do we have right now?

- ✅ AWS Account
- ✅ Budgeting Rules
- ✅ First, simple (stateless) Lambda function



AWS API Gateway



- create, publish, maintain and monitor **APIs**
- APIs are the **“front door”** for any application
- API Gateway can directly call AWS Lambda
- part of the **AWS Free Tier**: 1 Million calls per months (12 months free)

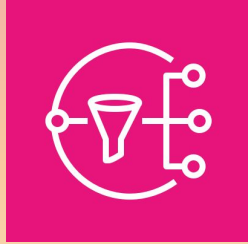
AWS Dynamo DB

- fast, flexible NoSQL database service
- “NoSQL” = no fixed schema, very flexible to various data
- part of the **AWS Free Tier**: 25 GB storage (always free)

AWS Event Bridge

- Create, trigger and manage **events**
- Make connections between different AWS services
- e.g. trigger AWS Lambda

AWS Simple Notification Service



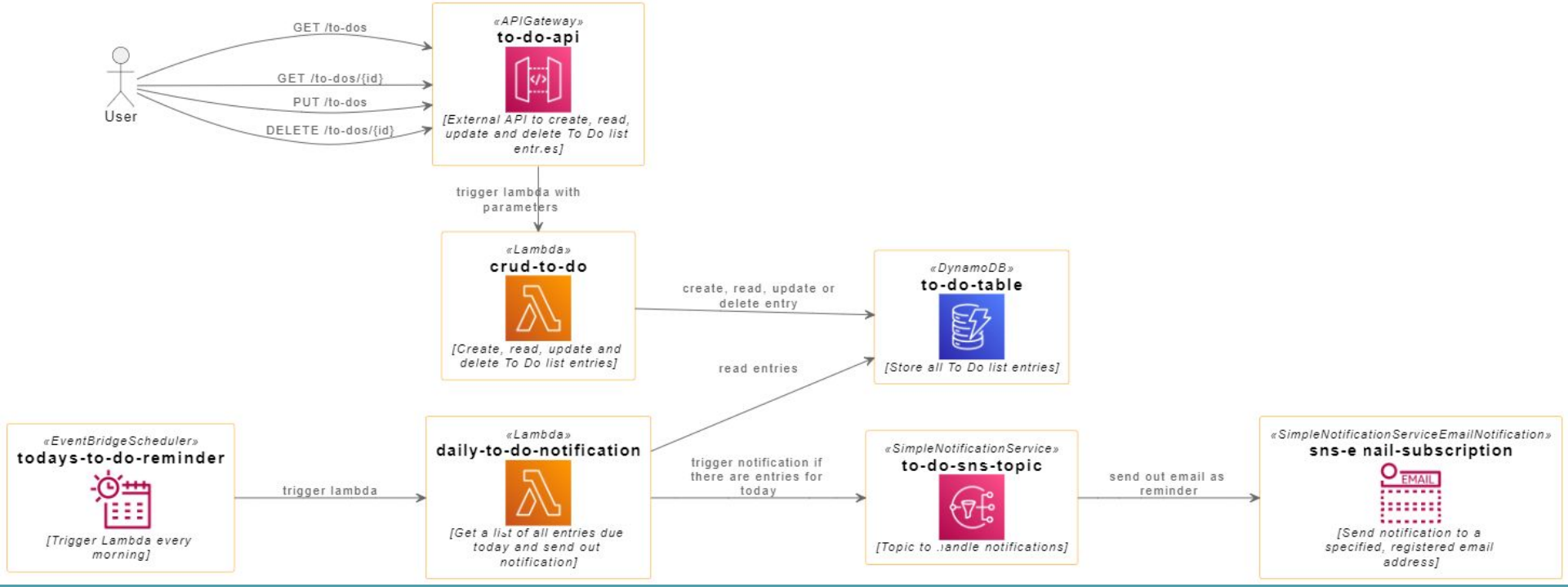
- Send notifications between applications or to people
- notifications to people: SMS, push notifications, emails

✍🏿 Practice - From a Lambda to an entire application

- 🎯 Create a **DynamoDB** table to store our To Do list
- 🎯 Build a **Lambda** that can create, read, update and delete To Do entries which are stored in a database
- 🎯 Build an **API Gateway** to make these functions publicly available

→ “how to”s and code for Lambdas

Review: What have we built?



Possible next steps

- Extend functionality
  - Support multiple users
  - Login / Security
  - Build a user interface
- Transform to “Infrastructure-as-Code”
  - Transform every “click” we made in the AWS Management Console into a written configuration → easy deployment, easier collaboration
- Build your own backend application on AWS 😊

Review / Closing

- ✅ What is the cloud? → service models, cloud providers
- ✅ Provisioned vs. “serverless” resources
- ✅ Introduction to AWS and the AWS Free Tier
- ✍🏽 Create a simple Lambda function
- ✍🏿 Build an To Do List app (using API Gateway, Lambda, DynamoDB)
- ✍🏼 Build a daily notification for our To Do List app (using EventBridge and Simple Notification Service)

Your workshop facilitator:
Iris Hunkeler