

Beginning AWS Lambda

Learn to write AWS Lambda Functions in Python

Noah Gift, UC Davis & Northwestern Lecturer (Cloud, ML, AI), Founder @Pragmatic AI Labs, Author: Pragmatic AI

Schedule

- **Part 1: Using Cloud9 to Develop Python Lambda Functions (90 min)**
- *QA (15 min)*
- *Break (15 min)*
- **Part 2: Creating Timed Lambdas (45 min)**
- *QA (10 min)*
- *Break (5 min)*
- **Part 3: Creating Event-Driven Lambdas (45 min)**
- *QA (15 min)*

Survey: Experience with AWS

- Novice (No experience)
- Beginner (< 1 Year)
- Intermediate (1-3 Years)
- Advanced (3+ Years)

Survey: Experience with Python

- Novice (No experience)
- Beginner (< 1 Year)
- Intermediate (1-3 Years)
- Advanced (3+ Years)

Part 1: Using Cloud9 to Develop Python Lambda Functions (90 min)

- Developing with Cloud9
- Launching Cloud9 and Workspace Configuration
- Creating and Deploying Lambda functions
- Importing Lambda functions
- Invoking Lambda functions
- Invoking Lambda function inside API Gateway

The screenshot displays the AWS Cloud9 IDE interface. The main editor shows a JavaScript file named `index.js` with the following code:

```
90 };
91
92 const handlersp = {
93   'LaunchRequest': function () {
94     this.emit('GetFact');
95   },
96   'GetNewFactIntent': function () {
97     this.emit('GetFact');
98   },
99
100 },
101 'GetFact': function () {
102   // Get a random space fact from the space facts list
103   // Use this.t() to get corresponding language data
104   const factArr = this.t('FACTS');
105   const factIndex = Math.floor(Math.random() * factArr.length);
106   const randomFact = factArr[factIndex];
107
108   // Create speech output
109   const speechOutput = this.t('GET_FACT_MESSAGE') + randomFact;
110   this.emit(':tellWithCard', speechOutput, this.t('SKILL_NAME'),
111
112 },
113 'AMAZON.HelpIntent': function () {
114   const speechOutput = this.t('HELP_MESSAGE');
115   const reprompt = this.t('HELP_MESSAGE');
116   this.emit(':ask', speechOutput, reprompt);
117 }
```

The right sidebar shows the 'Environment Members' section with a list of users: 'You (online)', 'aaron (online)', and 'rob (online)'. Below this is a 'Group Chat' window showing a conversation:

- You** (8 minutes ago): Hey Aaron, can you jump in here quick and look at these variables?
- aaron** (7 minutes ago): Sure, looking now
- aaron** (5 minutes ago): Ok, I've fixed the variables. Let's test it
- You** (5 minutes ago): thanks, before testing i want to show it to Rob real quick
- rob** (2 minutes ago): Looks ok. I don't see my Star trek facts though 😊

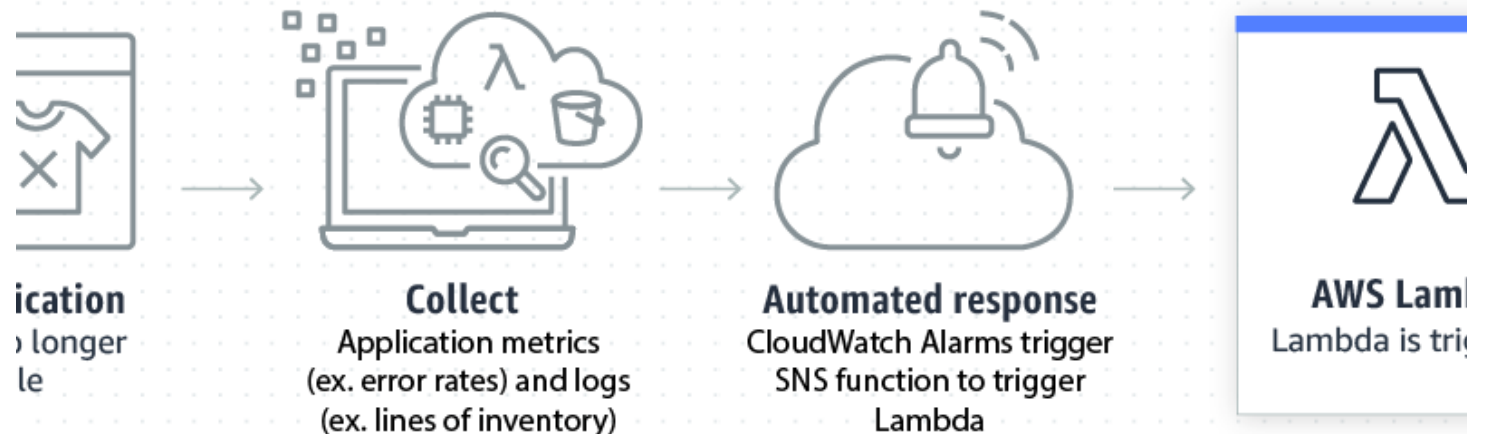
Blue lines connect the user avatars to their respective code snippets: 'aaron' to line 94, 'rob' to line 106, and 'claire' to line 114.

QA & Break Part 1

- QA (15 min)
Break (15 min)

Part 2: Creating Timed Lambdas (45 min)

- Using AWS Lambda with Cloudwatch Events
- Using AWS Lambda to populate AWS SQS (Simple Queuing Service)
- Using AWS Cloudwatch logging with AWS Lambda



QA and Break Part 2

- **QA (10 min)**
Break (5 min)

Part 3: Creating Event-Driven Lambdas (45 min)

Triggering

Triggering AWS
Lambda with AWS
SQS Events

Reading

Reading AWS SQS
Events from AWS
Lambda

Writing

Writing results to
AWS DynamoDB

Related Safari Properties

- [Pragmatic AI \(Book\)](#)
- [Essential Machine Learning and AI \(Video\)](#)
- [AWS Certified Machine Learning-Specialty \(Video\)](#)
- [Essential Machine Learning and Pragmatic AI \(Learning Path\)](#)
- Python for Data Science (Video)-Coming Soon
- AWS Certified Big Data-Speciality (Video)-Coming Soon

