# Lambda Function

AWS lambda allows you to execute code for any type of application

We can run code in response to certain events from other services like lex, dynamodb.

Supports number of programming languages like Node js, java, c# etc. Here we used python.

The Lambda function is used to integrate many AWS services. Here we choose the Lambda for two major reasons:

* data-processing for AWS services such as Amazon DynamoDB.
* Return and access responses to and from Amazon lex Libraries used in Lambda function are:
* **boto3**: Boto is the Amazon Web Services (AWS) SDK for Python. It enables Python developers to create, configure, and manage AWS services.
* **json:** Json is used here as all the data transmission in aws is done in json format.
* **smtplib: It creates a** session object that can be used to send mail to any internet machine with an SMTP

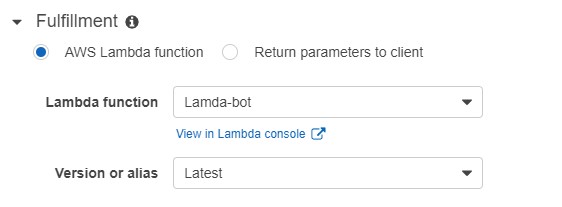
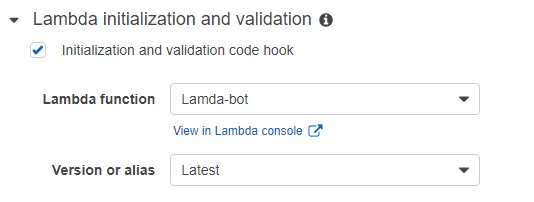
Now for the intents:

**Greetings:** Lambda function retrieve all the information entered by the user and from dynamodb all the details of the doctors related to the department chosen by the user are extracted and returned all the doctor details as a response card to lex.

**BookAnAppointment:** The lambda function is used for purpose to show the response card with the available timings of the selected doctor. Hence in this intent lambda function has an integral function of controlling the flow of the dialogue and also for error handling.

**Confirm:** The lambda function is used for purpose to book an appointment for the selected doctor and give a booking response to the patient via mail. Hence in this intent lambda function has an integral function of controlling the flow of the dialogue and also for error handling.

Enable the Lambda initialization and validation, Fulfillment with the created lambda function for each and every intent.



**Sample Snippet of the Code:**



Flexibilities with AWS Lambda:

* When you invoke a function, you can choose to invoke it synchronously or asynchronously. With synchronous invocation , you wait for the function to process the event and return a response.
* With asynchronous invocation, Lambda queues the event for processing and returns a response immediately.
* Scalability—According to users request it scales up or down the capacity of the function.
* Concurrency — ensure that a function can scale without fluctuations at simultaneous executions for the function

Working of Lambda function:

* The first time you invoke your function, AWS Lambda creates an instance of the function and runs its handler method to process the event.
* When the function returns a response, it stays active and waits to process additional events Ø If you invoke the function again while the first event is being processed, Lambda initializes another instance, and the function processes the two events concurrently.
* As more events come in, Lambda routes them to available instances and creates new instances as needed.
* When the number of requests decreases, Lambda stops unused instances to free up scaling capacity for other functions.