



Name: Tushar Panchal

En.No: 21162101014

Sub: EADC (Enterprise Application Development for Cloud)

Branch: CBA

Batch:61

-----PRACTICAL 07-----

❖ Question :

Imagine you're developing a distributed application spanning across IBM Cloud and AWS. You want to ensure seamless communication between components deployed on both platforms, particularly for notifying about events or triggering actions based on certain conditions.

The Event Notification Service (ENS) on IBM Cloud and the Simple Notification Service (SNS) on AWS are both messaging services designed to facilitate real-time communication and event-driven workflows within their respective cloud ecosystems.

Setting up IBM Cloud ENS:

Create an IBM Cloud account if you haven't already.

Access the IBM Cloud dashboard and navigate to the Event Notification Service.

Define your event sources and configure event rules to specify the conditions triggering notifications.

Generate necessary credentials or access tokens for authentication.

Subscribe your application endpoints (e.g., HTTP endpoints, messaging queues) to receive notifications.

Configuring AWS SNS:

Sign in to the AWS Management Console.

Navigate to the SNS service.

Create a new topic or select an existing one to which notifications will be published.

Define subscriptions, specifying endpoints (e.g., email addresses, HTTP/HTTPS endpoints, Lambda

functions) that should receive notifications.

Configure access policies to ensure proper authentication and authorization.

Obtain necessary credentials (e.g., access keys, IAM roles) for authentication when accessing

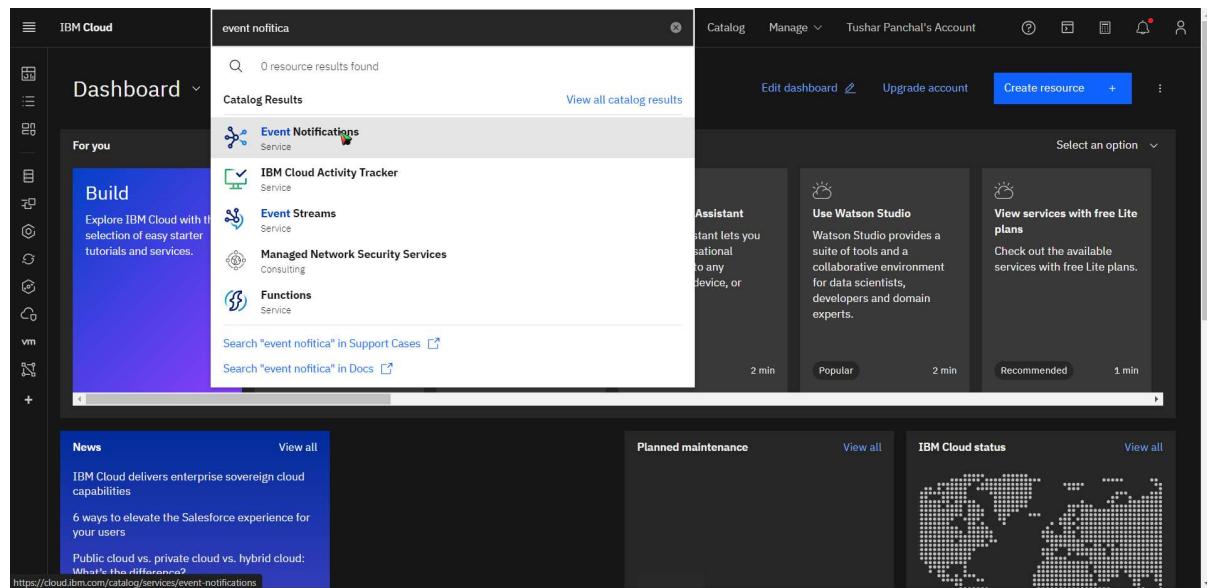
SNS APIs.

Integrate your application with SNS using AWS SDKs or RESTful APIs.

TASK: Integrate Slack as a destination with the event notification service of IBM Cloud to collect the events ingested by the sources to your Slack channel

» **Access the IBM Cloud dashboard and navigate to the Event Notification Service.**

» **Step 1:** navigate the service name event notification.



➤ Step 2: I created service by default name.

That's it event notification service created successfully

➤ Step 3: Then go to source and enable lifecycle option.

The screenshot shows the IBM Cloud Event Notifications interface. On the left, a sidebar lists options like Overview, Sources (which is selected), Destinations, Topics, Subscriptions, Templates, Integrations, SDKs, Service credentials, and Plan. The main area is titled 'Event Notifications-0w' and shows an 'Active' status with an 'Add tags' button. A message box in the top right corner says 'Source enabled' for the 'IBM Cloud Resource Lifecycle Events' source. Below this, a table lists a single source: 'IBM Cloud Resource Lifecycle Events'. The table columns include Name, Description, Type, Topics, and Enable. The 'Enable' column shows a green switch with the text 'Enabled' next to it. At the bottom of the table, there are pagination controls for 'Items per page: 10' and '1-1 of 1 item'.

➤ Step 4: navigate service name and Create cloud service storage.

Open another IBM cloud tab and serch for object storage

The screenshot shows the IBM Cloud Catalog interface. The search bar at the top contains the text 'object storage'. Below the search bar, a message says '0 resource results found'. Under the heading 'Catalog Results', there is a card for 'Object Storage Service'. To the right of the catalog results, there are 'Details' and 'Actions...' buttons. Below the catalog, there is another screenshot of the IBM Cloud interface showing the 'Cloud Object Storage' service details. The summary section includes 'Cloud Object Storage', 'Region: Global', 'Plan: Lite', 'Service name: Cloud Object Storage-fz', and 'Resource group: Default'. The 'Create' button is highlighted in blue. At the bottom, there is a 'Pricing' section with a 'Lite' plan listed, and a 'Create' button.

Name it Then simply hit create.

» Step 5: Then create a bucket in object storage tab.

The screenshot shows the 'Create bucket' page in the IBM Cloud Object Storage interface. On the left, there's a sidebar with 'Cloud Object Storage' selected. The main area displays four cards:

- Create a Custom Bucket**: Selected. Description: "Create a bucket by selecting bucket configurations that meet your object storage needs." Buttons: 'Create' (blue) and '→'.
- Quickly get started**: Description: "Create a Smart Tier storage class bucket in a region close to you and a service credential to connect your application." Buttons: 'Create' (grey) and '→'.
- Archive your data**: Description: "Create a Smart Tier storage class bucket in a region close to you with an archive rule and a service credential to connect your application." Buttons: 'Create' (grey) and '→'.
- Host a static website**: Description: "Create a Smart Tier storage class bucket in a region close to you with static web hosting configuration and a service credential to connect your application." Buttons: 'Create' (grey) and '→'.

Open another IBM cloud tab and search for object storage

Hit create a custom bucket and name it

The screenshot shows the 'Create custom bucket' page. The 'Unique bucket name' input field is highlighted with a red border and contains the value 'tusharpractical7th'.

Then select cross region and default location

The screenshot shows the 'Create bucket' configuration page. It includes the following sections:

- Resiliency**: Shows 'Cross Region' (Highest availability) selected. Note: "Resiliency cannot be modified after provisioning".
- Location**: Shows 'United States Geo (us-west-2)' selected. Note: "Location cannot be modified after provisioning".
- Storage class**: Shows 'Smart Tier (New!)' selected. Description: "Smart Tier automatically gives you the lowest storage rate based on your monthly activity." Note: "Storage class based on how often you expect to read the stored data. Pricing varies for each option."

At the bottom right is a large blue 'Create bucket' button.

» **Step 6:** Then left author option default and create bucket.

Service integrations (optional)

Encryption

Key management disabled

Encrypt the data in your bucket. Select a key management service. Key management can only be suspended after enabling.

Monitoring & Activity tracking

Activity tracking disabled

Increase operational visibility into the performance and health of your applications, services, and platforms.

Monitoring disabled

Monitor the activity of your IBM Cloud account. Troubleshoot errors, identify issues, and get alerts with real-time event logs.

Then hit create bucket.

As you can see here our bucket has been created.

IBM Cloud

Cloud Object Storage

Instances / Cloud Object Storage-fz / tusharpractical7th

Objects Configuration Permissions

If you're seeing more usage than expected, versions count towards your usage or you may have incomplete uploads [Learn more](#)

Prefix filter Object name

Size Last modified

Actions... Upload

» **Step 7:** Go into endpoint and copy public link.

IBM Cloud

Cloud Object Storage

Endpoints

Endpoints are used with your credentials (Bucket name, API Key, SDK) to tell your service where to look for your bucket. Choose an endpoint URL that is located in the same region as your service or application. [Learn more](#)

Select resiliency: Cross Region | Select location: United States Geo (us-geo)

	Public ⓘ	Private ⓘ	Direct ⓘ
us-geo	s3.us.cloud-object-storage.appdomain.cloud	s3.private.us.cloud-object-storage.appdomain.cloud	s3.direct.us.cloud-object-storage.appdomain.cloud
Dallas	s3.dal.us.cloud-object-storage.appdomain.cloud	s3.private.dal.us.cloud-object-storage.appdomain.cloud	s3.direct.dal.us.cloud-object-storage.appdomain.cloud
Washington	s3.wdc.us.cloud-object-storage.appdomain.cloud	s3.private.wdc.us.cloud-object-storage.appdomain.cloud	s3.direct.wdc.us.cloud-object-storage.appdomain.cloud
San Jose	s3.sjc.us.cloud-object-storage.appdomain.cloud	s3.private.sjc.us.cloud-object-storage.appdomain.cloud	s3.direct.sjc.us.cloud-object-storage.appdomain.cloud

Legacy Endpoints

s3.us.cloud-object-storage.appdomain.cloud

» **Step 8 :** Open event notification that we has been created before and open integrations and hit connect.

The screenshot shows the IBM Cloud Event Notifications interface. On the left, the navigation menu is visible with 'Integrations' selected. In the center, under the 'Integrations' section, there is a card for 'Collect Failed Events'. It displays the 'Connected instance' as 'Cloud Object Storage-fz', the 'Bucket name' as 'tusharpractical7th', and the 'Endpoint' as 's3.us.cloud-object-storage.appdomain.cloud'. A 'Connect' button is located at the bottom of this card. A modal window titled 'Collect Failed Events' is open on the right, prompting the user to find a COS bucket to store failed events. It contains fields for 'Instance name' (set to 'Cloud Object Storage-fz'), 'Bucket name' (set to 'tusharpractical7th'), and 'Endpoint' (set to 's3.us.cloud-object-storage.appdomain.cloud'). Buttons for 'Cancel' and 'Save' are at the bottom of the modal.

Then enter your bucket name that you have created and endpoint that you have copied and save

as you can see here we successfully configured public endpoint.

The screenshot shows the IBM Cloud Event Notifications interface. The 'Integrations' section is displayed, showing the 'Collect Failed Events' integration with its status set to 'Active' and the 'Bucket name' field containing 'tusharpractical7th'.

» **Step 9 :** Then go to destination & enable SMS & Email service.

The screenshot shows the IBM Cloud Event Notifications interface. The 'Destinations' section is selected in the navigation menu. It lists two destinations: 'IBM Cloud SMS service' and 'IBM Cloud Email service'. Both destinations have their 'Collect failed events' toggle switch set to 'On'. A success message is displayed in a toast notification: 'Success Destination IBM Cloud Email service is turned ON for collecting failed events.'

» Step 10: go in topic and create topic name & then create.

The screenshot shows the 'Create a topic' dialog box. In the 'Name' field, 'Practical_7' is entered. Under 'Sources', 'IBM Cloud Resource Lifecycle Events' is selected. The 'Event type (Optional)' dropdown is set to 'Instance'. The 'Event subtype (Optional)' dropdown has the value '\$.data.severity == 'LOW' && \$.* == 'any''. At the bottom right, a blue 'Create' button is visible.

Then don't select event subtype and event type then hit create

As you can see here our topic has been created

The screenshot shows the 'Topics' section of the interface. A message box at the top right says 'Topic created' with the note 'Practical_7 topic is added to the list.' Below it, the 'Practical_7' topic is listed in the table with its source as 'IBM Cloud Resource Lifecycle Events' and 0 subscriptions.

» Step 11: go to subscription & name it & select topic that we has been created & choose SMS service in destination.

The screenshot shows the 'Create a subscription' dialog box. In the 'Name' field, 'EADC_Practical_7' is entered. Under 'Topic', 'Practical_7' is selected. Under 'Destination', 'IBM Cloud SMS service' is selected. At the bottom right, a blue 'Create' button is visible.

then enter your phone number.

The screenshot shows the IBM Cloud interface for managing subscriptions. In the top navigation bar, 'IBM Cloud' is selected, and the search bar contains 'Practical_7'. The main area is titled 'Event Notifications-0w' and shows a 'Subscriptions' section. A sub-section titled 'Recipients' explains that up to 3 phone numbers can be added. Below this, there are tabs for 'Invited (0)', 'Active (0)', and 'Unsubscribed (0)'. The 'Invited (0)' tab is active. A text input field contains '+919687834849'. An 'Add +' button is visible next to it. Below the input field, a note says 'Enter phone numbers, separated by comma. Numbers should begin with + and country code. Refer Learn more for supported countries.' At the bottom of the page are 'Cancel' and 'Create' buttons, with 'Create' being highlighted in blue.

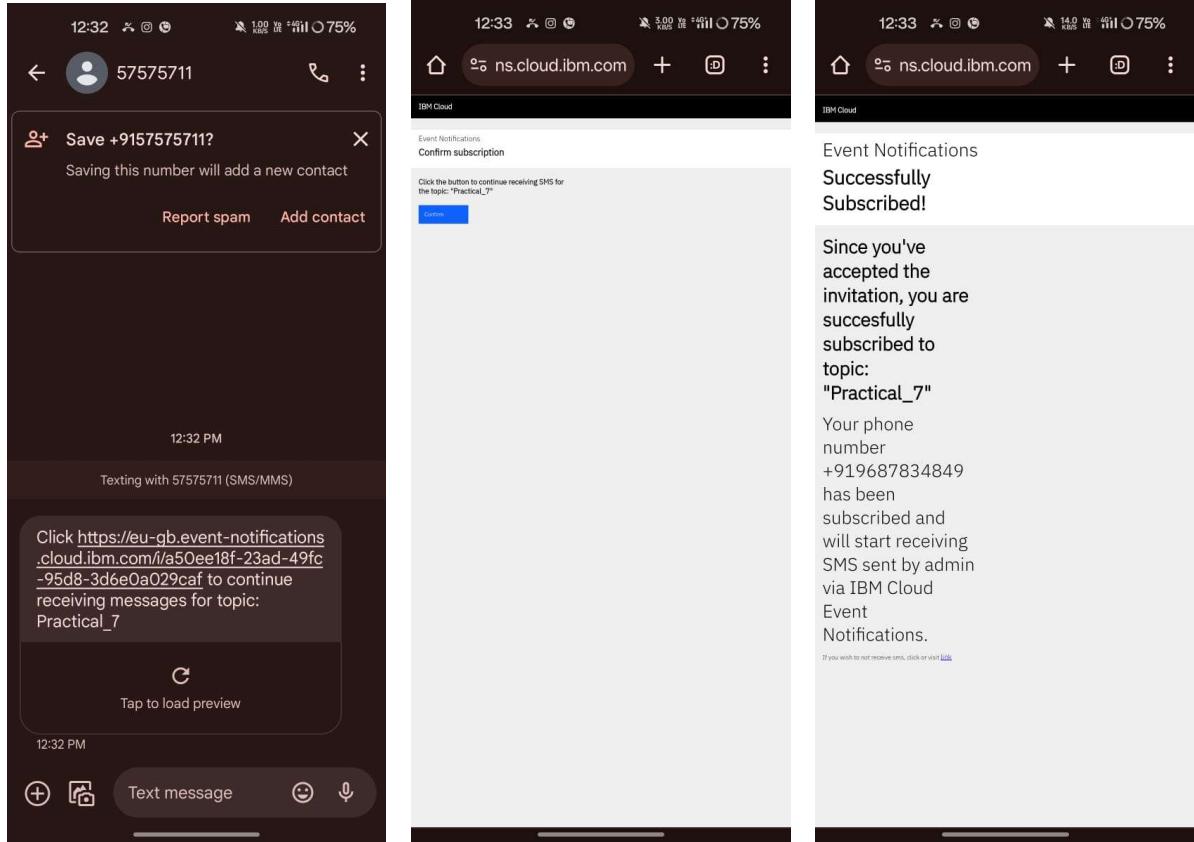
As you can see here my phone number has been added

This screenshot shows a list of phone numbers under the 'Phone numbers' heading. The list contains one item: '+919687834849'. To the right of the list is a column labeled 'Date invited' with a single dash '-' below it. At the bottom of the list, there are pagination controls: 'Items per page: 10' and '1–1 of 1 item'.

You can see here my subscription has been created successfully.

This screenshot shows the 'Event Notifications-0w' resource list. The left sidebar has 'Subscriptions' selected. The main area displays a table for 'Subscriptions'. The table has columns: 'Name', 'Description', 'Topic', 'Destination', and 'Type'. One row is listed: 'EADC_Practical_7', 'Practical_7', 'IBM Cloud SMS service', 'IBM SMS'. At the bottom of the table are pagination controls: 'Items per page: 10' and '1–1 of 1 item'.

After that you will receive sms on your mobile number and open that link on your smartphone & confirm that to subscribe service



» Step 12 : create a new topic for mail name it and hit create.

I named it Tushar_Mail and selected IBM cloud Resource Lifecycle Events and hit create :

The screenshot shows the 'Create a topic' dialog in the IBM Cloud Event Notifications interface. The 'Name' field is filled with 'Tushar_Mail'. The 'Source' dropdown is set to 'IBM Cloud Resource Lifecycle Events'. The 'Event type (Optional)' dropdown is set to 'Select a type'. The 'Advanced conditions -JSON path (Optional)' field contains the condition '\$.data.severity == 'LOW' && \$.* == 'any''. The 'Create' button is highlighted in blue at the bottom right.

As you can see here my topic created successfully :

The screenshot shows the IBM Cloud Event Notifications interface. On the left, a sidebar lists options like Overview, Sources, Destinations, Topics, Subscriptions, Templates, Integrations, SDKs, Service credentials, and Plan. The Topics option is selected. The main area displays a table of topics. A success message in a toast notification says "Topic created Tushar_Mail topic is added to the list." The table has columns for Name, Description, Sources, and Subscriptions. It shows two entries: "Practical_7" with 1 subscription and "Tushar_Mail" with 0 subscriptions.

That's it now go to Subscription under overview of event notification that we created before.

Go to subscription & hit create then name it and select topic that we created for mail before and select Email service in Destination:

The screenshot shows the "Create a subscription" dialog box. In the "Subscription details" section, the "Name" field contains "EMAIL TEST". The "Topic" dropdown is set to "Tushar_Mail". The "Destination" dropdown is set to "IBM Cloud Email service". The "Add notification payload" checkbox is unchecked. The "Reply to" section is partially visible at the bottom. A large blue "Create" button is at the bottom right of the dialog.

Add reply to & give mail id to get notifications of activities and add valid email address in below of it :

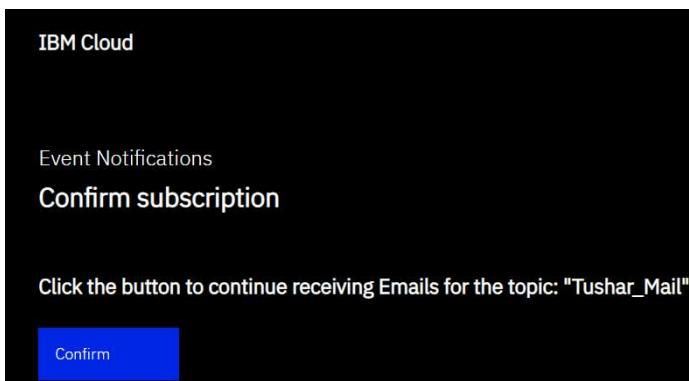
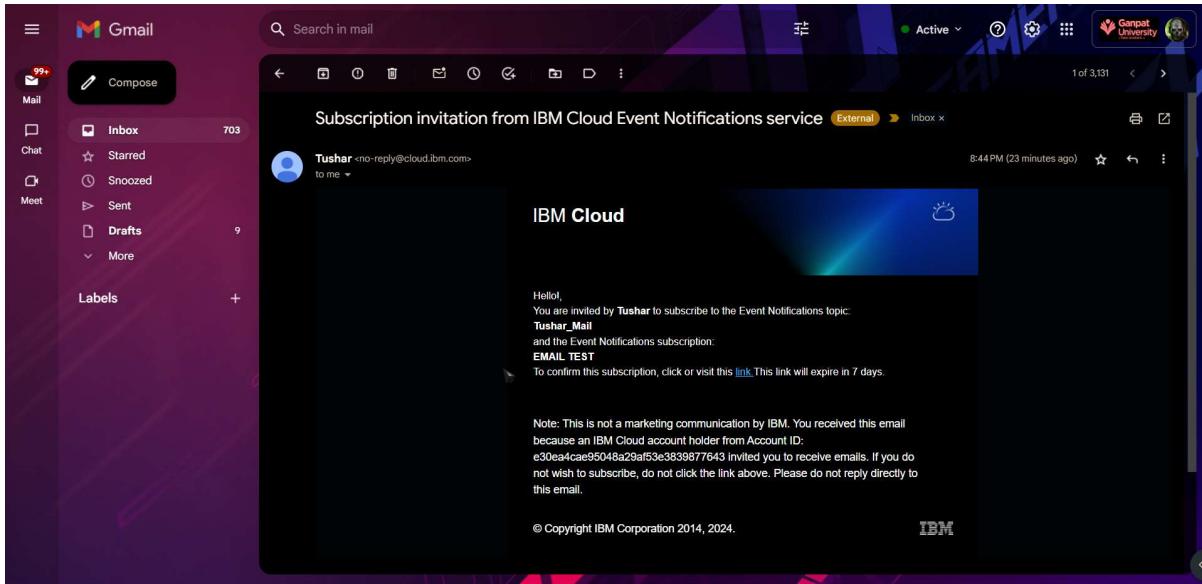
The screenshot shows the configuration of a new subscription. The 'Topic' is set to 'Tushar_Mail' and the 'Destination' is 'IBM Cloud Email service'. A checkbox for 'Add notification payload' is unchecked. The 'Name' field contains 'Tushar'. In the 'Reply to' section, the 'Name' is 'Tushar Panchal' and the 'Email' is 'tusharpanchal21@gnu.ac.in'. Below this, the 'Invited (1)' tab is selected, showing one user invited but not yet accepted. The user's email address is listed in the 'Email addresses' input field.

Email address	Date invited	Invitation expiry date
tusharpanchal21@gnu.ac.in	3/15/2024, 8:44:31 PM	3/22/2024, 8:44:31 PM

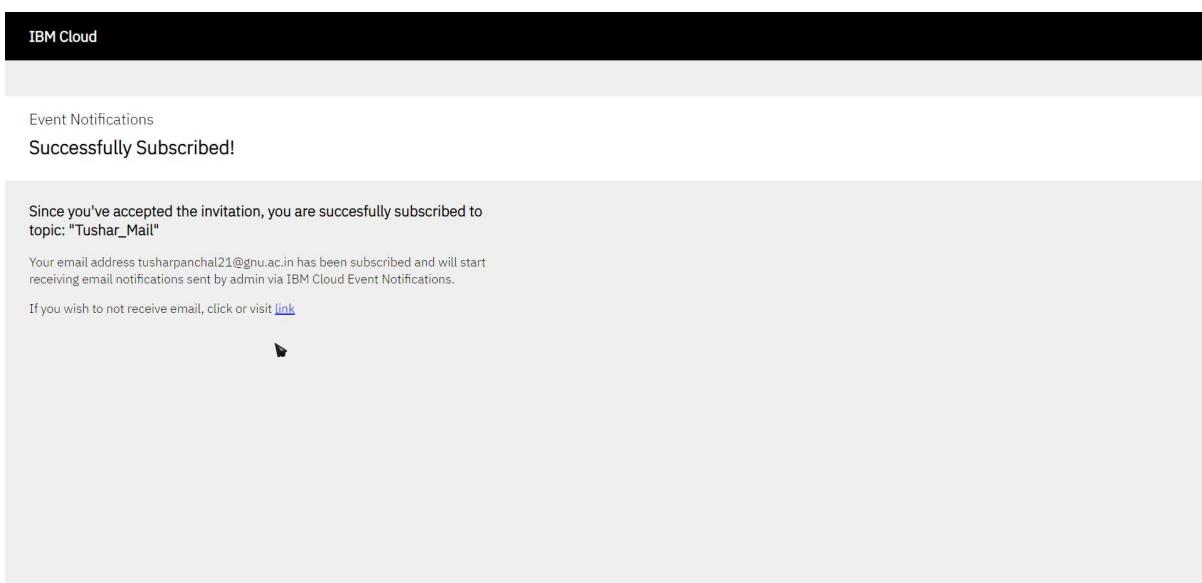
As you can see here my subscription has been created successfully:

The screenshot shows the list of existing subscriptions. The 'Subscriptions' tab is selected. There are two entries: 'EADC_Practical_7' (Topic: Practical_7, Destination: IBM Cloud SMS service, Type: IBM SMS) and 'EMAIL TEST' (Topic: Tushar_Mail, Destination: IBM Cloud Email service, Type: IBM Email). A success message in a toast notification states 'Subscription created EMAIL TEST subscription added to the list.' The left sidebar shows other options like Overview, Sources, Destinations, Topics, Templates, Integrations, SDKs, Service credentials, and Plan.

» **Step 13:** Check subscription invitation on email hit link to subscribe service.



As you can see service successfully subscribed.



» **Step 14:** To check that we receive notification or not make a speech-to-text service.

Directly search for speech to text in catalog seach box

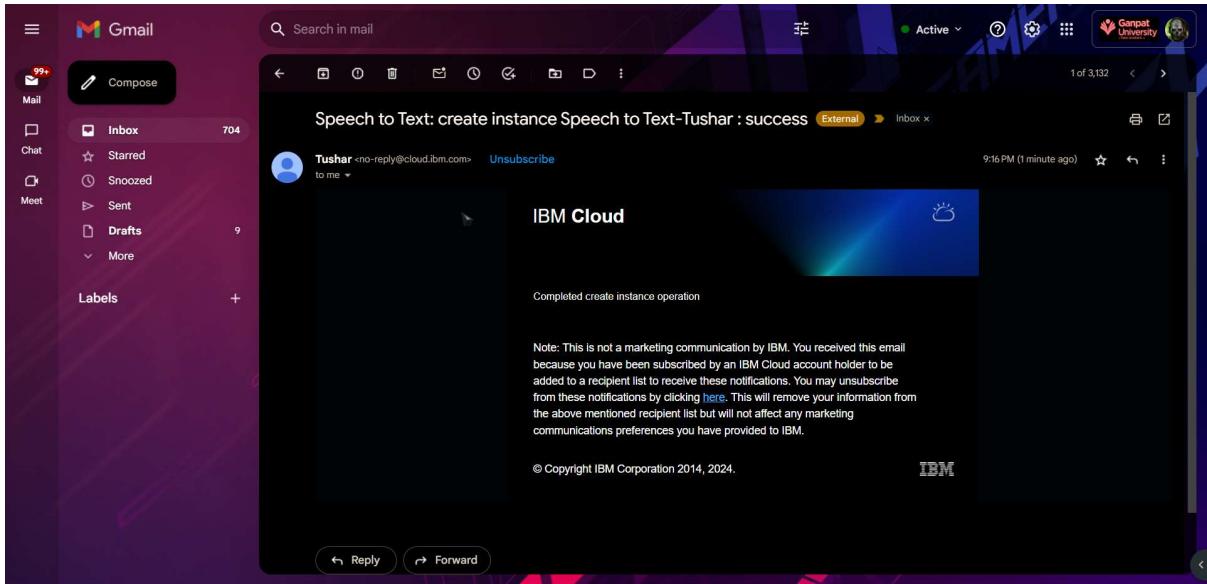
The screenshot shows the IBM Cloud Catalog interface. On the left, there's a sidebar with service type (Service), provider (IBM), last updated (03/04/2024), category (AI / Machine Learning), compliance (EU Supported, HIPAA Enabled, IAM-enabled), location (Sydney, Frankfurt, London, Tokyo, Washington DC, Dallas), and related links (API docs, Docs, Terms). The main content area has a title 'Speech to Text' with a subtitle 'Low-latency, streaming transcription'. It features tabs for 'Create' and 'About'. A dropdown for 'Select a location' is set to 'Sydney (au-syd)'. Below it, a section for 'Select a pricing plan' shows the 'Lite' plan with '500 Minutes per Month' and 'Free'. A note states: 'The Lite plan gets you started with 500 minutes per month at no cost. When you upgrade to a paid plan, you will get access to Customization capabilities.' Another note says: 'Lite plan services are deleted after 30 days of inactivity.' There's also a 'Plus - NEW!' plan with 'Minutes Per Month' and 'Simple Volume Tiers'. A link 'Click to view tiers and pricing detail' is present. On the right, a 'Summary' panel shows details: Service name: Speech to Text-mm, Location: Sydney, Plan: Lite, Resource group: Default. It also includes a checkbox for accepting terms and conditions, a 'Create' button, and an 'Add to estimate' button.

Then name it and create

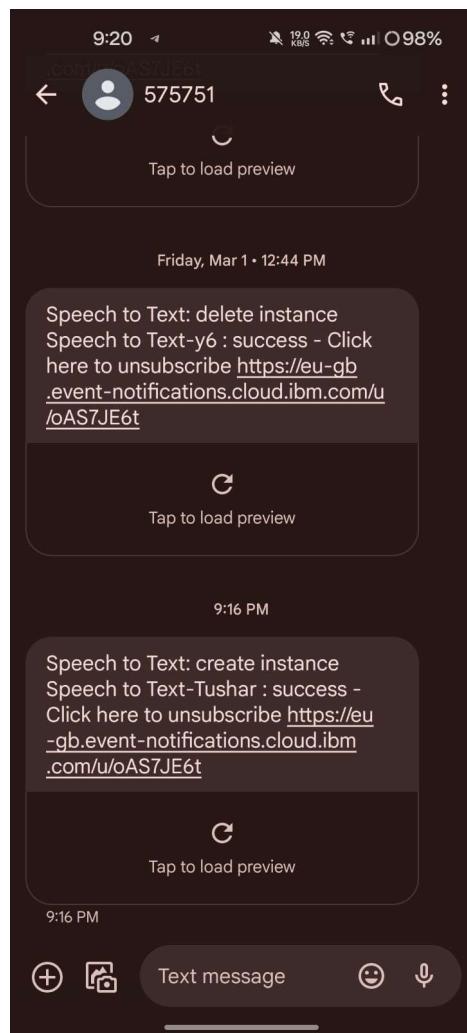
This screenshot shows the configuration step in the IBM Cloud Catalog. The left sidebar remains the same. The main area now displays configuration fields: 'Service name' (set to 'Speech to Text-Tushar'), 'Select a resource group' (set to 'Default'), 'Tags' (with example 'env:dev, version-1'), and 'Access management tags' (with example 'Examples: access:dev, proj:version-1'). The right side shows the same summary and creation UI as the previous screenshot, with the 'Create' button highlighted in blue.

That's it our speech to text service has been created successfully.

Now check on mail you will receive an email that you created and speech to text service



And on your mobile number you also receive SMS of that too



create a Continuous Delivery and select region London

Continuous Delivery

Support DevOps best practices by using Git, issue tracking, source code vulnerability analysis, and CI/CD pipelines in the Cloud.

Create **About**

Type: Service
Provider: IBM
Last updated: 03/07/2024
Category: Developer tools
Compliance: EU Supported, Financial Services, Validated, IAM-enabled
Location: Frankfurt, London, Washington DC, Dallas, Sydney, San Paulo

Select a location: London (eu-gb)

Select a pricing plan
Displayed prices do not include tax. Monthly prices shown are for country or location: United States

Plan	Features and capabilities	Pricing
Lite	Continuous Delivery for teams of up to 5 users 5 users per resource group per region 500 Delivery Pipeline classic jobs or Tekton steps run per resource group per month 500 MB of private Git Repos storage per resource group per region 30 days retention of Delivery Pipeline run records	Free

The Lite plan offers the full capabilities of Continuous Delivery, with usage limits, to small teams at no cost.

I have read and agree to the following license agreements:
[Terms](#)

Creating...

Add to estimate

create Tool chain and select the region London

Toolchains / Create a toolchain / Build your own toolchain

Create **About**

Toolchain name: empty-toolchain-20240315185230509

Select region: London

Select a resource group: Default

Cancel **Create**

ASK A QUESTION

Toolchains / empty-toolchain-20240315185230509 [Add tags](#)

Details **Actions...**

Overview

Your toolchain is ready! Quick start: You can now add tool integrations. For more details, view the documentation.

Search... [Add](#)

Repositories: 0 total Delivery pipelines: 0 total IBM Cloud tools: 0 total

ASK A QUESTION

After creating toolchain you have to add the Github As repositories

The screenshot shows the 'Configure GitHub' step in the IBM Cloud toolchain setup. The 'GitHub Server' dropdown is set to 'GitHub (https://github.com)'. The 'Auth type' dropdown is set to 'OAuth'. A message box states: 'You must authorize before you can configure this tool integration.' with a blue 'Authorize' button.

After Giving authorization to git hub you have to select the existing in repository type and select the your exiting repository

The screenshot shows the 'Configure GitHub' step after authorization. The 'GitHub Server' dropdown is set to 'GitHub (https://github.com)'. The 'Auth type' dropdown is set to 'OAuth'. The 'Repository type' dropdown is set to 'Existing'. The 'Repository URL' input field contains the URL 'https://github.com/Tushar007079/EADC_PRACTICAL_2'. Other fields include 'Git Integration Owner' set to 'Tushar007079', and checkboxes for 'Enable GitHub Issues' (checked) and 'Track deployment of code changes' (unchecked).

Now go to toolchain and add the Delivery Pipeline
Name it and hit create integration

The Delivery Pipeline service automates continuous deployment.

Choose between **Classic** pipeline that is defined through the UI, or the new **Tekton**-based 'pipeline as code' that is defined in a Git repository.

Name your pipeline, choose the way you will define it, and then select Create Integration. This will take you to the pipeline configuration page to finish setting it up.

IBM Financial Services Validated

View Docs

TOOLCHAIN empty-toolchain-20240315185230509

Create Integration

Let's add stage in our pipeline:

Tushar_Practical7 | Delivery Pipeline

Add Stage + Actions...

Tushar_Practical7 | Stage Configuration

MyStage

Input Jobs Workers Environment prop...

Input settings

Input type Git repository

Git repository EADC_PRICAL_2

Git URL https://github.com/Tushar007079/EADC_PRICAL_2.git

Branch main

Stage trigger Run job only when this stage is run manually

Actions...

Choose input type and git repository as above
In jobs add job as build

Then save it

As you can see our stage has been added

Same as add the other stage for testing take input as build artifacts.

In jobs select test in add job

Then save it

As you can see our stage has been added

after creating this two Stage access your git file and some changes and push it to git

Make some changes in file or code for github you selected in continuous delivery and commit it thorough terminal.

```

File Edit Selection View Go Run ...
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS POSTMAN CONSOLE SERIAL MONITOR

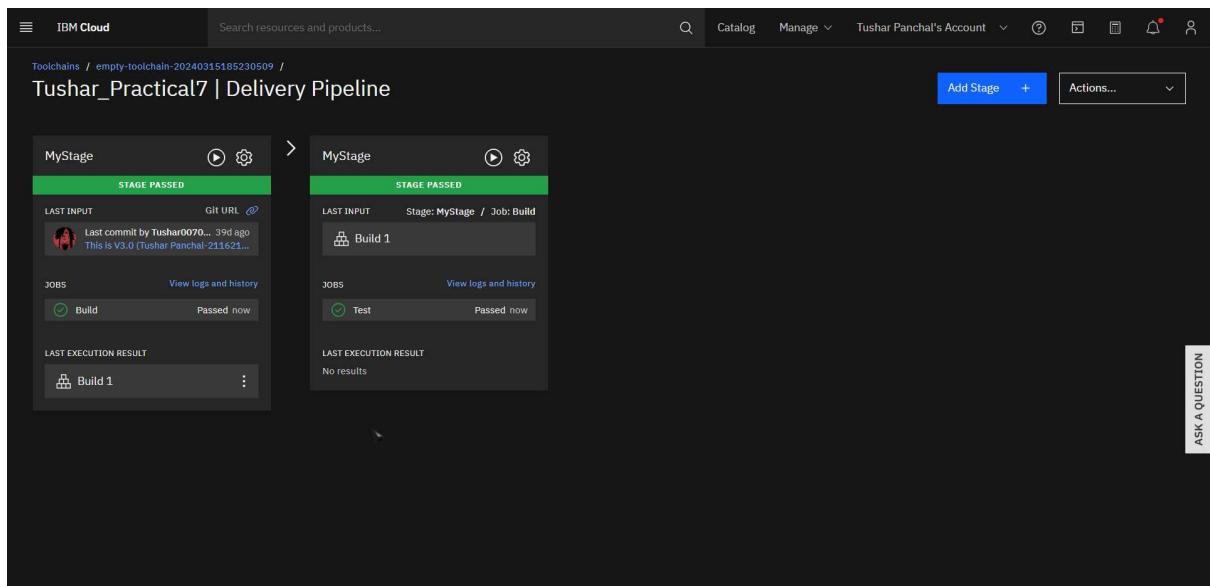
< > Practical-2
pwsh 20.10.0 91% 16:00:41
> git add .
pwsh 20.10.0 91% 16:00:41
> git commit -m "I'm tushar This is practical 7 demo"
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
pwsh 20.10.0 92% 16:00:42
> git push -u origin main --force
remote: Invalid username or password.
fatal: Authentication failed for 'https://github.com/Tushar007079/EADC_PRACTICAL_2.git'
pwsh 20.10.0 92% 16:00:42
> git push
info: please complete authentication in your browser...
Enumerating objects: 100%, 35/35, done.
Counting objects: 100%, 35/35, done.
Delta compression using up to 8 threads.
Compressing objects: 100%, 33/33, done.
Writing objects: 100%, 882.05 KiB | 29.40 MiB/s, done.
Total 33 (delta 25), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100%, 25/25, completed with 2 local objects.
To https://github.com/Tushar007079/EADC_PRACTICAL_2.git
  814a8c3..146cbc1 main -> main
pwsh 20.10.0 92% 16:00:43
> git push -u origin main --force
Everything up-to-date
branch 'main' set up to track 'origin/main'.
pwsh 20.10.0 92% 16:00:43
> git push
Everything up-to-date
pwsh 20.10.0 92% 16:00:43
>

```

Ln 12, Col 4 Spaces: 2 UTF-8 CRLF () JavaScript Go Live Colorize: 0 variables Colorize

As you can see I commit changes it automatically started stages



After that you have to add one more service of Event Notification in toolchain

For that hit add tool in toolchain & search for event notification
Configure and create it

IBM Cloud Search resources and products... Catalog Manage Tushar Panchal's Account ? ☰ 🔍 📁 📈 📉 📣 🚙 🚧 🚧 ASK A QUESTION

Toolchains / Toolchain details / Add tool Integration / Configure Event Notifications Create Integration

Use Event Notifications to deliver different notifications for your Toolchain and Pipeline events to various destinations. Before you can use the Event Notifications tool integration, you must create an instance of Event Notifications for this tool integration. For more information about Event Notifications, see the Event Notifications documentation.

Important: To deliver notifications, the Event Notifications instance must have an IAM authorization policy that allows this toolchain to send events to the selected Event Notifications service instance. Learn more.

IBM Financial Services Validated

View Docs

TOOLCHAIN empty-toolchain-20240315185230509

Name ⓘ event-notification-tushar

Event Notifications instance ⓘ Event Notifications-0w (6eb125d-716a-40a5-9f45-41f96edca8b)

Authorization ⓘ Create Authorization

Then hit create authorization

IBM Cloud Search resources and products... Catalog Manage Tushar Panchal's Account ? ☰ 🔍 📁 📈 📉 📣 🚙 🚧 🚧 ASK A QUESTION

Toolchains / Toolchain details / Configure Event Notifications Save Integration

Enter a name for this Event Notifications integration in your toolchain and select your Event Notifications instance from the list.

Name ⓘ event-notification-tushar

Event Notifications instance ⓘ Event Notifications-0w (6eb125d-716a-40a5-9f45-41f96edca8b)

Authorization ⓘ Could not create authorization Close

The authorization already exists.

IBM Cloud Search resources and products... Catalog Manage Tushar Panchal's Account ? ☰ 🔍 📁 📈 📉 📣 🚙 🚧 🚧 ASK A QUESTION

Toolchains / empty-toolchain-20240315185230509 Add tags ⚡ Details Actions...

Overview Q. Search... Add

Connections Manage

Repositories 1 total EADC_PRACTICAL_2 New GitHub https://github.com/Tushar007079/EAD...

Delivery pipelines 1 total Tushar_Practical7 New Delivery Pipeline Triggered 3/16/24, 12:43 AM

IBM Cloud tools 1 total event-notification-tushar New Event Notifications

Now we can see the toolchain in Sources of event notification. hit on Enable.

Name	Description	Type	Topics	Enable
IBM Cloud Resource Lifecycle Events	This is a pre-defined source which publishes resource lifecycle events from IBM Cloud.	resource-lifecycle-events	2	<input checked="" type="checkbox"/> Enabled
Toolchain Events (4da2a1f3-6428-444f-9dd7-16965f156fc0)	Events from the toolchain (4da2a1f3-6428-444f-9dd7-16965f156fc0)	toolchain	0	<input checked="" type="checkbox"/> Enabled

Now we need to create topic in Topics section. The topics can then be subscribed to get notifications

Name	Description	Sources	Subscriptions
Practical_7		IBM Cloud Resource Lifecycle Events	1
Tushar_Mail		IBM Cloud Resource Lifecycle Events	1
toolchain_topic		Toolchain Events (4da2a1f3-6428-444f-9dd7-16965f156fc0)	0

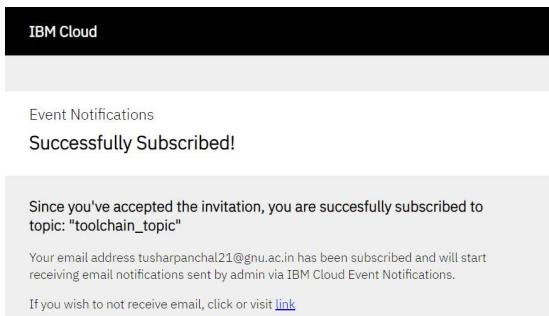
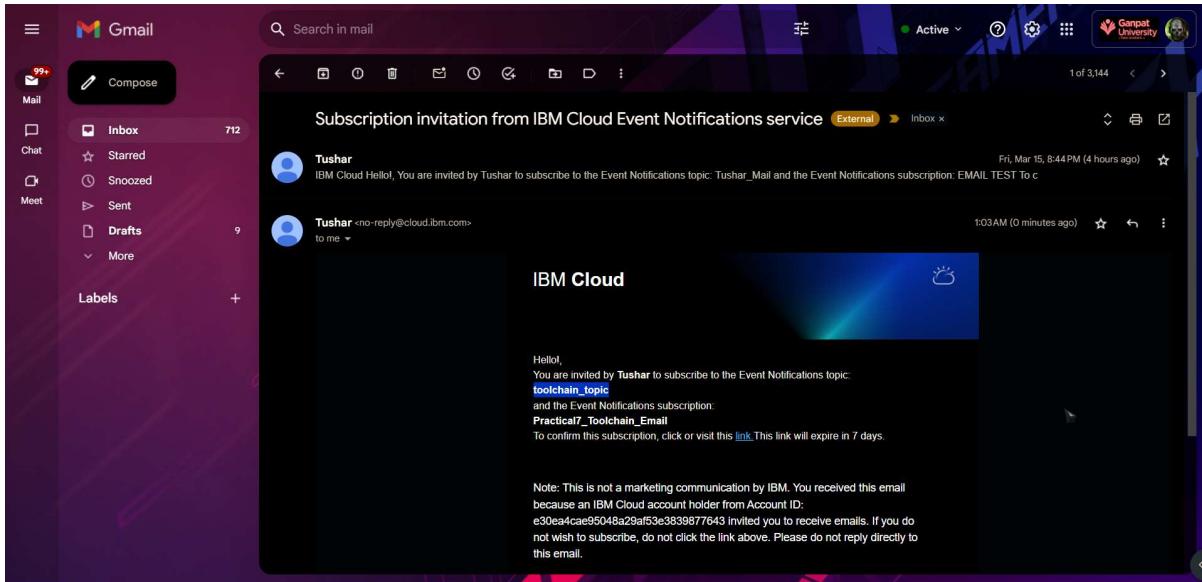
Create a new subscription select toolchain_topic and in destination enter email address:

The screenshot shows the 'Event Notifications-0w' resource list in the IBM Cloud interface. On the left, a sidebar lists various options like Overview, Sources, Destinations, Topics, Subscriptions, Templates, Integrations, SDKs, Service credentials, and Plan. The 'Subscriptions' option is selected. In the main panel, there's a search bar and a 'Create a subscription' button. The 'Subscription details' section asks for a name and description. The 'Name' field contains 'Practical7_Toolchain_Email'. The 'Topic' dropdown is set to 'toolchain_topic'. The 'Destination' dropdown is set to 'IBM Cloud Email service'. Below these, there's a checkbox for 'Add notification payload' which is unchecked. The 'Reply to' section shows 'Name' as 'Tushar' and 'Email' as 'tusharpanchal21@gnu.ac.in'. Under 'Recipients', there's a 'Create' button. The overall interface has a dark theme.

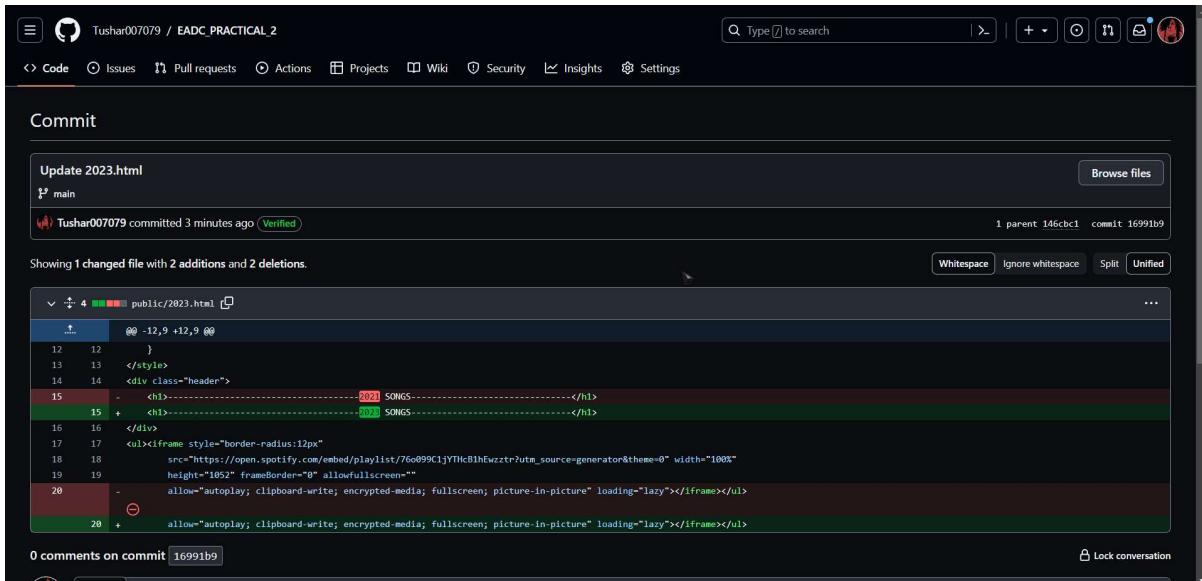
Now we need to give email which will send an confirmation email to get subscription after we create the subscription. Right now the email is in invited section .

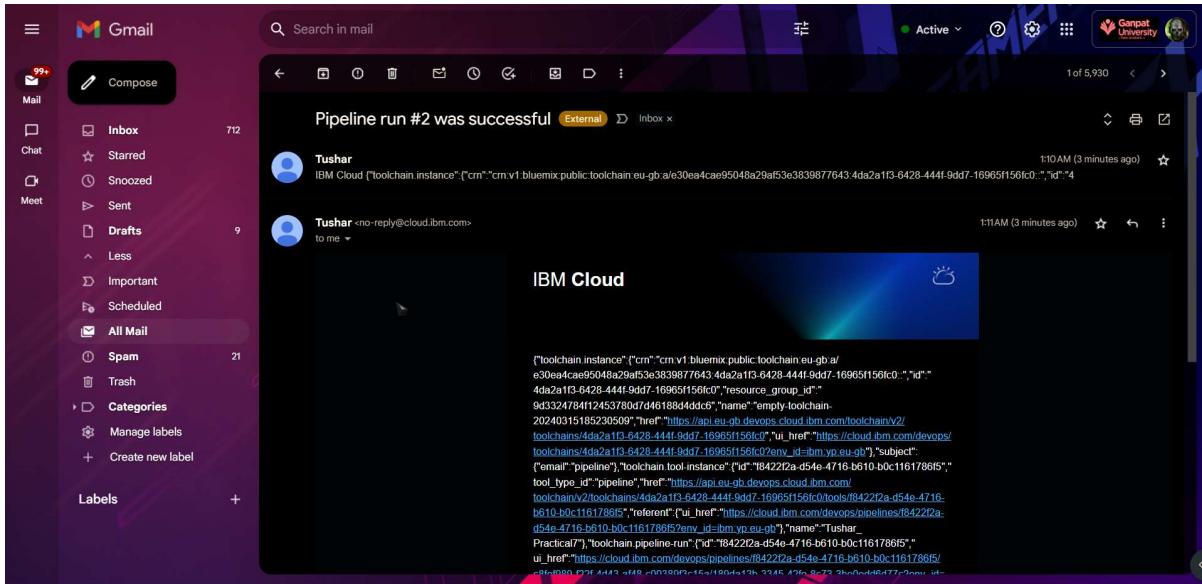
This screenshot shows the same 'Event Notifications-0w' interface, but the focus is on the 'Recipients' section. It displays a table with one row under the 'Invited' tab. The row shows 'Name' as 'Tushar Panchal' and 'Email' as 'tusharpanchal21@gnu.ac.in'. Below this table, there's a section for adding more email addresses with a text input field 'Enter valid email addresses' and a 'Create' button. The interface maintains its dark theme.

Then you will receive mail to subscribe notification
Confirm that by link



Now we can check the notification service by changing the code in the repository that is connected with the toolchain pipeline.

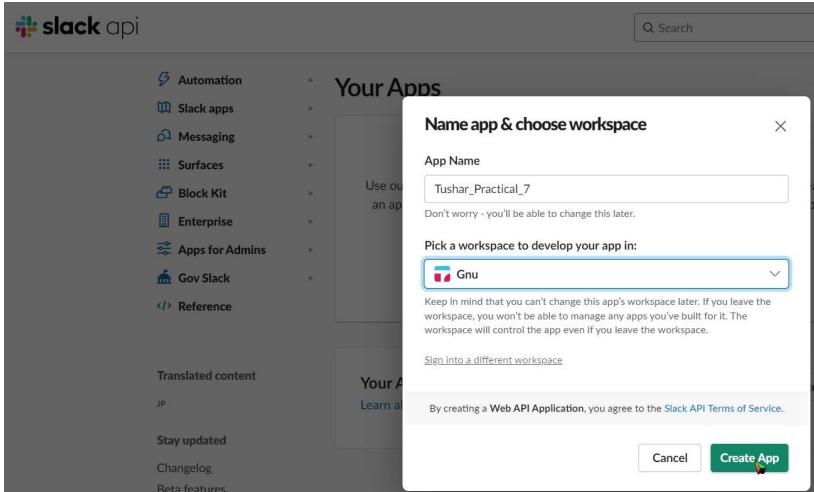




Integrating with slack. Open api.slack.com and select create a new app

The screenshot shows the Slack API homepage. On the left, there's a sidebar with various API categories like Automation, Slack apps, Messaging, Surfaces, Block Kit, Enterprise, Apps for Admins, Gov Slack, and Reference. The main area has a heading 'Your Apps' with a sub-section 'Build something amazing.' It includes a 'Create an App' button. Below that is another section 'Your App Configuration Tokens' with a 'Generate Token' button. At the bottom, there's a 'Stay updated' section with links to Changelog, Beta features, Developer blog, Roadmap, and @SlackAPI.

name it and create a new workspace. We also have option to add an existing workspace



Now activate incoming webhooks which will allows to post messages in slack channels through external sources like IBM Cloud.

The screenshot shows the 'Incoming Webhooks' configuration page for the 'Tushar_Practical_7' app. On the left, a sidebar lists 'Settings' (Basic Information, Collaborators, Socket Mode, Install App, Manage Distribution), 'Features' (App Home, Org Level Apps), 'Incoming Webhooks' (selected), and 'Submit to App Directory'. The main content area is titled 'Activate Incoming Webhooks' with a toggle switch set to 'On'. It explains that incoming webhooks allow posting messages from external sources using normal HTTP requests with a JSON payload. It also notes that adding a bot user is required if none exists. Below this is a section for 'Webhook URLs for Your Workspace', which instructs users to dispatch messages with their webhook URL using a POST request. A sample curl command is provided: `curl -X POST -H "Content-Type: application/json" -d '{"text": "Hello, Slack!"}' https://hooks.slack.com/services/T00000000/B00000000/XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX`.

Select a channel or user to send notifications to

The screenshot shows a permission request dialog for the 'Tushar_Practical_7' app. It asks, 'Where should Tushar_Practical_7 post?' with a dropdown menu showing '# general'. Below the dropdown is a note: '# Tushar_Practical_7 requires a channel to post to as an app'. At the bottom are 'Cancel' and 'Allow' buttons.

Once successfully completed we will get a webhook URL, copy the URL. we also get a sample curl request if we want to work with APIs directly.

The screenshot shows the Slack Incoming Webhooks configuration page. On the left, there's a sidebar with various app categories like Org Level Apps, Interactivity & Shortcuts, Slash Commands, Workflow Steps, OAuth & Permissions, Event Subscriptions, User ID Translation, App Manifest, Beta Features, Submit to App Directory, Review & Submit, Give feedback, Slack Help, Contact, Policies, and Our Blog. The 'Incoming Webhooks' section is highlighted. The main content area has a heading 'Webhook URLs for Your Workspace' and a sample curl command. Below it is a table showing a single webhook entry:

Webhook URL	Channel	Added By
https://hooks.slack.com/services/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	#general	Tushar Panchal Mar 15, 2024

At the bottom, there's a 'Copy' button next to the URL and a 'Copy' button for the entire curl command. A 'Copy' button is also present in the table header row.

Open the toolchain>add>Slack and write the webhook URL, slack channel and team name. Now Create Integration. We can select here for which pipeline events we want the notifications for. For example if we want notifications for failed pipeline events only then we can only check the pipeline failed option.

The screenshot shows the IBM Cloud Toolchains interface for configuring a Slack integration. At the top, there's a navigation bar with 'IBM Cloud', a search bar, and account information. The main area is titled 'Configure Slack' and contains the following fields:

- Slack webhook: An input field with a placeholder '...', a copy icon, and a link icon.
- Slack channel: An input field with a placeholder 'random', a copy icon, and a link icon.
- Slack team name: An input field with a placeholder 'gnu-zfh9227', a copy icon, and a link icon.
- Automated Slack Notifications: A note stating 'Customize your selection by choosing the events for which you want to receive notifications. Notifications will automatically be sent to the specified slack channel only for the selected events.'
- Pipeline events: A list with checkboxes for 'Pipeline started', 'Pipeline succeeded', and 'Pipeline failed', all of which are checked.
- Toolchain events: A list with checkboxes for 'Tool added' and 'Tool removed', both of which are checked.

On the right side, there's a 'Save Integration' button and a 'ASK A QUESTION' link.

Here we can see in 3rd party tools the channel #random which we selected in slack

The screenshot shows the IBM Cloud interface for the toolchain 'empty-toolchain-20240315185230509'. The top navigation bar includes 'Catalog', 'Manage', 'Tushar Panchal's Account', and various icons. The main area displays sections for 'Overview', 'Connections', 'Manage', 'Repositories' (1 total, EADC_PRactical_2), 'Delivery pipelines' (1 total, Tushar_Practical_7, triggered 3/16/24, 1:10 AM), 'IBM Cloud tools' (1 total, event-notification-tushar), and 'Third-party tools' (1 total, #random). A search bar at the top says 'Search resources and products...'.

Message in the slack channel

The screenshot shows the Slack interface for the '#general' channel. The left sidebar shows channels like '# general', '# notification', etc. A message from 'Tushar_Practical_7' at 1:34 AM states: 'added an integration to this channel: Tushar_Practical_7'. Below it, another message from 'Tushar_Practical_7' at 1:39 AM says: 'Webhook configured by DevOps Services integration (462af241-ab2d-43be-90f5-bf7c91ead234) Service Slack '#random' has been bound to toolchain empty-toolchain-20240315185230509 by tusharpanchal21@gnu.ac.in and is configured Webhook reconfigured by DevOps Services integration (462af241-ab2d-43be-90f5-bf7c91ead234)'.

The message shows the stage, status of the job and also when it started, how long it took and also some other information.

The screenshot shows the Slack interface for the '#general' channel. The left sidebar shows channels like '# general', '# notification', etc. A message from 'Tushar_Practical_7' at 1:43 AM provides detailed logs of a pipeline run: 'Stage 'MyStage' #3 has STARTED Triggered by tusharpanchal21@gnu.ac.in Input: https://github.com/Tushar007079/EADC_PRactical_2.git [main] Started: Saturday, March 16th 1:45:49 AM'. Below it, another message from 'Tushar_Practical_7' at 1:43 AM says: 'Job 'Build' in Stage 'MyStage' #3 has PASSED Triggered by tusharpanchal21@gnu.ac.in Started: Saturday, March 16th 1:46:00 AM Duration: 10 seconds'. Further down, a message from 'Tushar_Practical_7' at 1:46 AM says: 'Stage 'MyStage' #3 has PASSED Triggered by tusharpanchal21@gnu.ac.in Input: https://github.com/Tushar007079/EADC_PRactical_2.git [main] Started: Saturday, March 16th 1:45:49 AM Duration: 20 seconds'. At the bottom, a message from 'Tushar_Practical_7' at 1:46 AM says: '1:46 Stage 'MyStage' #3 has STARTED Triggered by pipeline Input: Build 3 Started: Saturday, March 16th 1:46:14 AM'.

➡️ Configuring AWS SNS (For Email).

Sign in to the AWS Management console.

Here I'm using Praveen's account because my account has been suspended because of payment issues....

» Step 1: Navigate to the SNS service.

The screenshot shows the AWS Management Console interface. The top navigation bar includes the AWS logo, a search bar with 'SNS', and a user dropdown for 'praveen singh kushwah'. Below the search bar, the sidebar lists various services like Features, Resources, Documentation, Knowledge Articles, Marketplace, Blogs, Events, and Tutorials. The main content area displays search results for 'SNS' under 'Services' and 'Features'. The 'Simple Notification Service' is the top result under 'Services', described as 'SNS managed message topics for Pub/Sub'. Other listed services include Amazon Pinpoint SMS, Route 53 Resolver, and Route 53. Under 'Features', there are sections for 'Events' and 'Hosted zones', each with a single item listed.

» Step 2: Name it and create a topic.

The screenshot shows the 'Amazon Simple Notification Service' landing page. It features a banner about message archiving and replay for FIFO topics. The main heading is 'Amazon Simple Notification Service: Pub/sub messaging for microservices and serverless applications.' Below this, there's a detailed description of what Amazon SNS offers. To the right, a modal window titled 'Create topic' is open, showing a text input field with the value 'EADC_PRAC7_TUSHAR'. There are 'Next step' and 'Start with an overview' buttons at the bottom of the modal. At the bottom of the page, there's a 'Benefits and features' section with two items: 'Reliably deliver messages with...' and 'Automatically scale your...'. The footer contains standard AWS links for CloudShell, Feedback, and legal information.

➤ Step 3: Select standard in type and give display name.

The screenshot shows the 'Create topic' interface in the AWS SNS console. The 'Type' dropdown is set to 'Standard'. The 'Name' field is filled with 'EADC_PRAC7_TUSHAR'. The 'Display name' field is filled with 'Tushar'. The 'Create topic' button is highlighted in orange at the bottom right.

➤ Step 4: then hit create topic.

The screenshot shows the confirmation message for creating the topic. The topic details are displayed: Name (EADC_PRAC7_TUSHAR), Display name (Tushar), ARN (arn:aws:sns:ap-south-1:851725560915:EADC_PRAC7_TUSHAR), and Type (Standard). The 'Create subscription' button is highlighted in orange at the bottom.

» **Step 5:** Hit create subscription button.

The screenshot shows the AWS SNS 'Topics' page. On the left, there's a sidebar with options like Dashboard, Topics (which is selected), Subscriptions, Mobile (Push notifications, Text messaging (SMS), Origination numbers), and CloudWatch Metrics. The main area is titled 'Details' for the topic 'EADC_PRAC7_TUSHAR'. It shows the Name (EADC_PRAC7_TUSHAR), Display name (Tushar), ARN (arn:aws:sns:ap-south-1:851725560915:EADC_PRAC7_TUSHAR), Topic owner (851725560915), and Type (Standard). Below this, the 'Subscriptions' tab is selected, showing 'Subscriptions (0)'. There are buttons for Edit, Delete, Request confirmation, Confirm subscription, and Create subscription. A search bar and filters for ID, Endpoint, Status, and Protocol are present. A message states 'No subscriptions found' and 'You don't have any subscriptions to this topic.' At the bottom, there's a 'Create subscription' button. The footer includes links for CloudShell, Feedback, and various AWS terms like Privacy, Terms, and Cookie preferences.

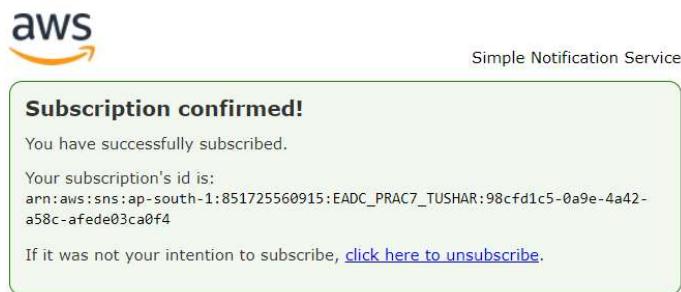
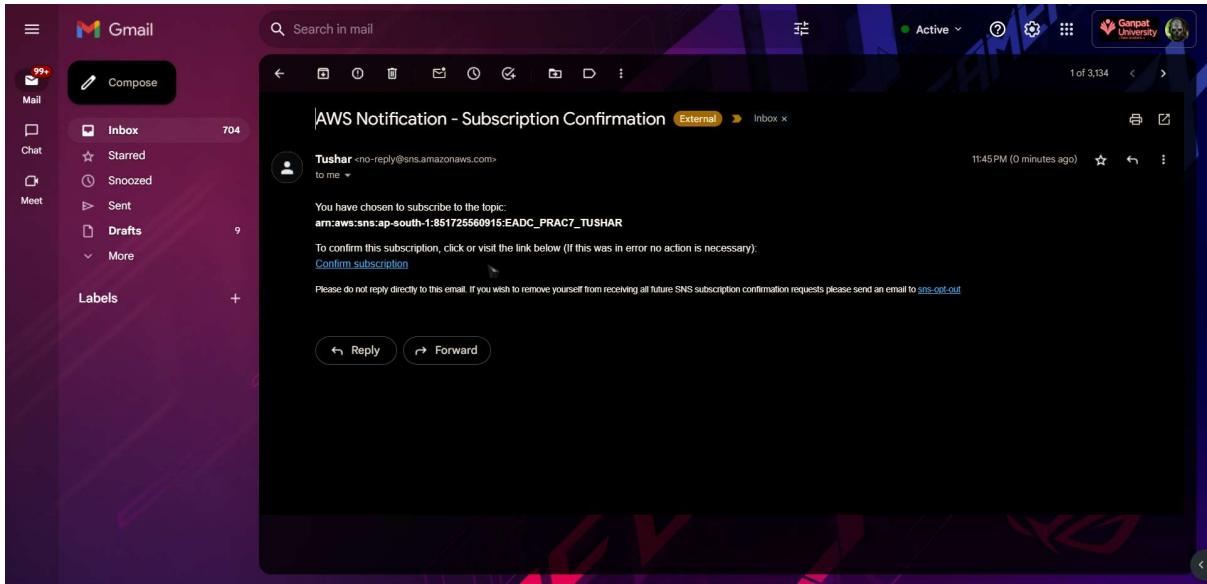
» **Step 6:** Give endpoint as your mail id to get notification and track activities Left other as default than create.

Select email in protocol

And in endpoint enter your email to get notifications

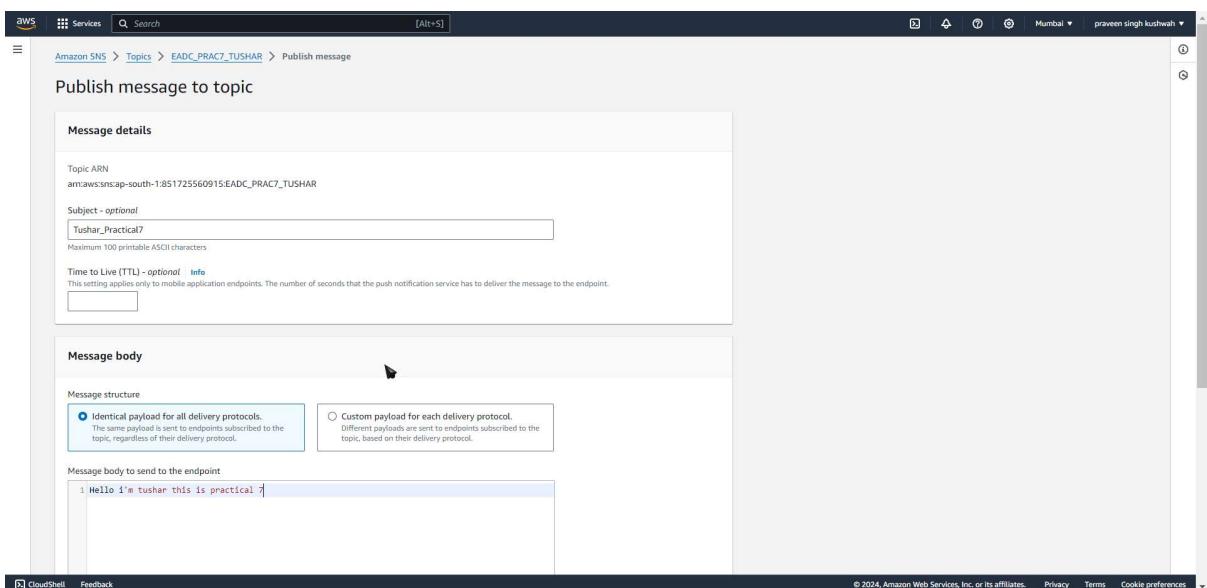
The screenshot shows the 'Create subscription' wizard. Step 1: Details. It shows the Topic ARN (arn:aws:sns:ap-south-1:851725560915:EADC_PRAC7_TUSHAR), Protocol (Email), and the Endpoint (tusharpanchal21@gnu.ac.in). A note says 'After your subscription is created, you must confirm it.' Step 2: Subscription filter policy - optional. Step 3: Redrive policy (dead-letter queue) - optional. At the bottom, there are 'Cancel' and 'Create subscription' buttons. The footer includes links for CloudShell, Feedback, and various AWS terms like Privacy, Terms, and Cookie preferences.

» **Step 7:** we need to subscribe service from mail to get notifications.

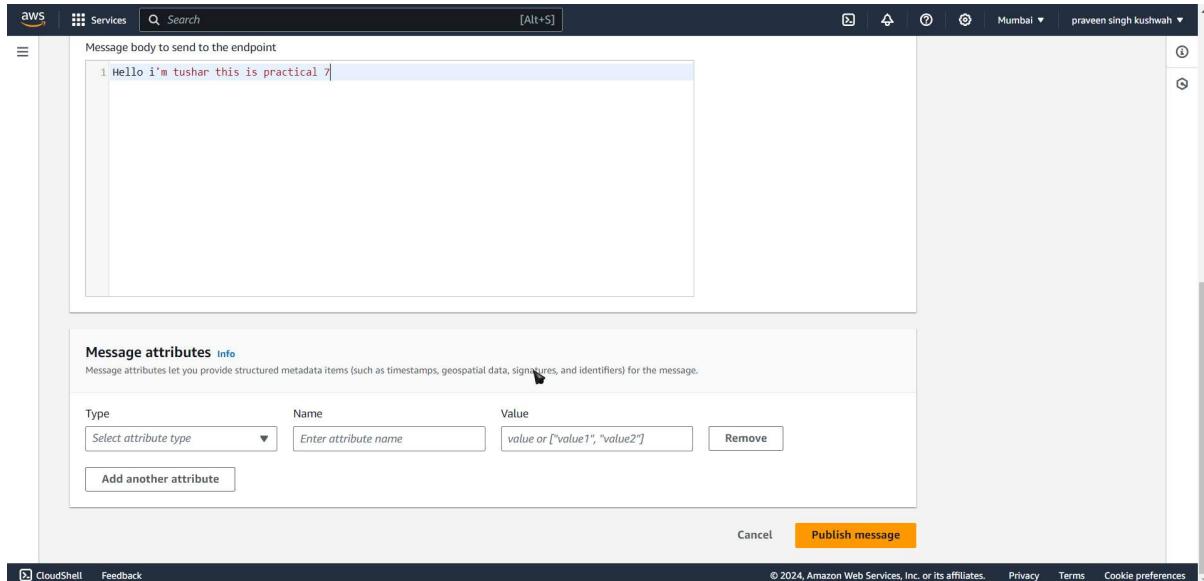


» **Step 8:** go to publish message to topic and name it Select identical for all delivery protocols.

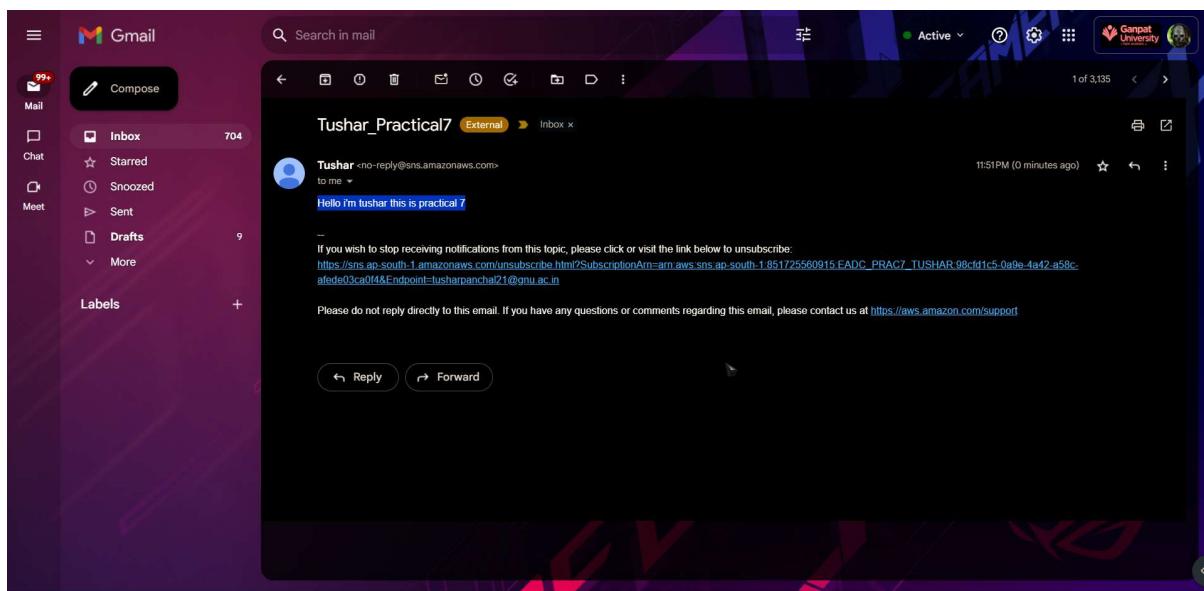
Add message in body to send to the endpoint



» Step 9: publish the message that we added.



As you can see here I got notified that message in email



For some reason I can't add phone number in subscription otherwise I show in practical that how to configure AWS SNS for SMS